SEGA® COMPUTER

NOVEMBER ISSUE 1984

The Official Sega User Club Magazine

REVIEWS

Latest Software Review
Cassette Software
Education

FEATURES

Beginners Programming
Conversion Chart
Sega Sound
Results for Sega

GAMES PROGRAMS

Fireman
Doll Program
Jingle Bells
Indy GP
Graphics Demonstration

INTRODUCTION

Dear Member

Christmas time is always a busy time of the year. Peace on earth and good will to all men is the thought in everyones mind, and it is a time for giving and sharing.

Computers will without doubt be on the list of many people and undoubtedly make an excellent gift with long lasting benefits for the lucky recipient. Computer owners will also no doubt be hoping for software rather than soft toys to fill out their Christmas stocking.

To help you through the software jungle, bearing this in mind, we have published a complete software list plus pictures where possible, giving age recommendations and explanations of the programs, so you can save yourself the time and your feet the agony of sorting through the shelves. Our stockists should carry a full range of programs once you have made your selection.

The long awaited Sega control station will also be on the shelves and we feature the final version of its extended basic in this issue to tempt those enthusiasts who will be interested. One word of advice to these people, please order early, as there are limited pre Christmas computers and our stockists will only receive one delivery.

November is always a very busy month in the computer world, not only because of the Christmas build up, but also the numerous consumer and trade fairs which abound the two major arenas being Christchurch B & B and Auckland C.E.S. From our point of view we look on these with mixed feelings. The benefits from being able to meet so many people in such a short period can be enormous and usually it is a great exercise in getting to know people who are our users, and learning from them the ways in which we can increase their enjoyment. Sales to potential customers are also greatly increased by our efforts.

The sheer volume of people and the hours of the shows can create problems, however, namely communication (at the end of the show it is usually a cross between semafor and telepathy) as the voices are worn as to be almost inaudable. The feeling of togetherness amongst the Sega demonstrators is always increased by the end of each show, as without each other's support we would never be able to make it up the last set of stairs to the bar each night (purely mechanical of course).

Our short story for the magazine comes from Mr Bill Dewhurst, and takes us back some 10 years to a small town in Lancashire, England, where he was at that time living and working as a Television repair man for a large rental company.

His companions were well known practical jokers, and they kept wary eyes open for one anothers' pranks. So when Bill received a call from a concerned elderly lady to say that each time her T.V. set was switched on the fish in the pond began behaving erractically, and her father got electric shocks in his potting shed, he was naturally suspicious.

On arriving at the house he was immediately even more suspicious to find the lady was well into her sixties, however her father was still alive and quite able (although relunctant) to make the journey to and from his potting shed.

The television was turned on and Bill was instructed to wait for 15 minutes as it usually took that long for the fish to react, the time passed while he drank tea and awaited the howls of laughter from his triumphant companions, who he was sure were hiding somewhere.

The 15 minutes up, Bill was ushered in to the garden to view the fishpond, where to his amazement, the Goldfish were behaving in an extremely agitated manner and swimming rapidly around in circles.

The father of the house was then dragged bodily from his armchair, still in his slippers, complaining vigourously as his lungs would allow that he did not want to go into the shed. His daughter continued to reassure him by saying "the man has to see what happens," but this did not appear to set his mind at ease, as the prospect of several volts coursing through his body loomed even nearer.

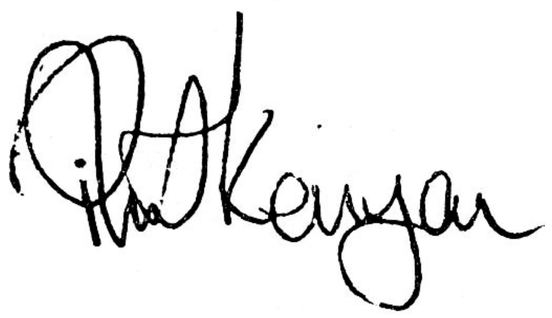
Sure enough the unfortunate man touced the outside of the garden shed, his eyes lit up and he fled to the safety of his armchair with a speed which was astonishing for his years.

By now Bill was convinced no-one could have thought of and staged anything so bizarre as this, and set about finding the cause of the mysterious happenings. Once the back of the set was removed, the first clue emerged, being a fairly old set, the earth was not fixed to the chassis, but taken through an earth socket in the wall. On tracing the wire he discovered it led out through the garden, past the fish pond which was cracked and leaking and finally attached to the frame of the garden shed, whose wooden structure was damp and beginning to rot.

As water conducts electrical currents, the fish were receiving a mild current causing their agitation, and each time Grandad touched the shed the current went to earth through the wet wood, through the gaping hole in his slippers.

All in a days work for a T.V. repair man!

Yours faithfully,
P. Kenyon
GRANDSTAND LEISURE LTD



Contents

Read Letters 2
The Light Dawneth 3
New Venture for Grandstand 4
Program Dissection 5
Printer Demonstration 8
Beginners Programming 10
Addendum 12
Fireman 13
Conversion Chart 16
Doll Program 17
Sega Sound 19
Sound Demonstrations 20
Why Grandstand Get Results for Sega 21
Sega Superstar 23
Jingle Bells 24
Software Review 25
Indy GP 27
Graphics 29
Cassette Software Education 30
F Basic 32

LOCAL SEGA USERS CONTACTS

PUKEKOHE SEGA USERS CLUB
C/o 4 Roose Avenue
Pukekohe
Contact: Selwyn
Ph. Pukekohe 86-583

AUCKLAND CENTRAL
c/o 287 Broadway Furniture
Newmarket
Contact: George Shaw
Ph. 547-543

ROTORUA
Rotorua Sega Users Club
C/- 61 Devon Street
ROTORUA

TOKOROA
Tokoroa Sega Users Group
C/- 1 Pio Pio Place
TOKOROA
Contact Geoff Phone Number 67105
Tokoroa

TARANAKI
South Taranaki Microcomputer Society
D. M. Beale
7A Clive Street
HAWERA

NAPIER
Napier Sega Users Club
Sec E. P. Lins
41 Higgins Street
NAPIER

The above are contact names and addresses for Sega Users Clubs. If you wish to have your club advertised write to Sega Users Club P.O. Box 2353, Auckland

READER'S LETTERS

DEAR EDITOR

Thank you for the September issue of the Sega Users Club Magazine. Would you please answer the following questions for me:

1. I am using the Basic Level II cartridge and am unable to create CIRCLES on my computer.
2. I am unable to obtain any sprites as soon as I enter PATTERN I get:

"Syntax Error"

3. Your programs frequently use I and 1 which to me can be confusing. Instead of using I, is it possible to use another letter which cannot be mistaken for 1.
4. In your first issue you mention a number of ways in which the computer can be damaged by switching on and off. I have been doing this, but have no way of telling whether I have caused any damage. Is there a way I check the computers functions.

EDITORS REPLY

Unfortunately as you have a IIA cartridge you are not able to create circles or sprites due to the restricted memory available. To run the cassette based programs and most of the programs in the magazine, you will require a IIIA or IIIB cartridge.

We are now printing all the programs in the magazine on the Dot Matrix printer. This should avoid confusion between "1's" and "I's," and we shall be attempting wherever possible to avoid the use of "I" as a variable.

With regards to your queries on switching the computer "on" and "off." The section in the first magazine was merely to advise people to be careful with the machines when doing so. You should always make sure when you remove a cartridge that you turn the machine off, as that too may damage the machine.

You have most probably not damaged your computer in any way as the Sega is a very robust machine. You will soon be able to tell if you have done any damage to your machine, as it will not work. There is a self test program built in to the unit itself that produces a number of beeps when the computer is first turned on.

If this is happening turn your computer off and reinsert the cartridge. If this is of no help we suggest you send your computer to Grandstand where it can be serviced fully.

DEAR EDITOR

I have found two mistakes in your second issue of the listing of the "Candy Kid," Maze Chase Game.

The first is simple
Line 39Ø should read:

```
390 FOR X=26 TO 154 STEP 64:
FOR Y=21 TO 149 STEP 32:
GOSUB 490:NEXT Y:NEXT X
```

The second mistake is in line 42Ø. It should read.

```
420 RESTORE 780
```

Yours sincerely
Mike Densem

EDITORS REPLY

Thank you for pointing this out. Please except our apologies.

DEAR EDITOR

I have recently read that you can upgrade the 16K basic cartridge to 32K of RAM by the addition of two extra 4416 chips. If this is so, how much would it cost and where could I buy the chips.

Robert Telfer
Tauranga

EDITORS REPLY

Our Computer Technician tells me that it is possible to upgrade a 16K basic cartridge to 32K of RAM with two 4416 chips and two 1Ø4 16V ceramic caps. Grandstand does not stock these parts so will not be performing the conversion. The chips cost about $4Ø-5Ø each and are available from larger electronic shops and wholesalers.

DEAR EDITOR

I have been able to get a good colour monitor but I cannot get a cable to hook it up to my Sega. Could you please supply me with the pin connections so that I can make one myself.

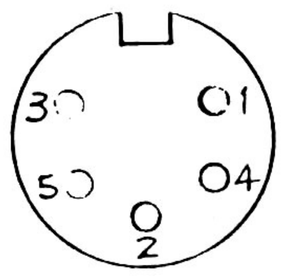
J Morris
Auckland

P.S. Great magazine. Keep up the good work.

EDITORS REPLY

Thank you for your support for the club magazine. Listed below is the information required to get your colour monitor working.

```
Video terminal
PAL  : 5 PIN DIN JACK
```



Pin No.	Signals
1	AUDIO-OUT
2	GND
3	COMPOSITE-OUT
4	GND
5	GND

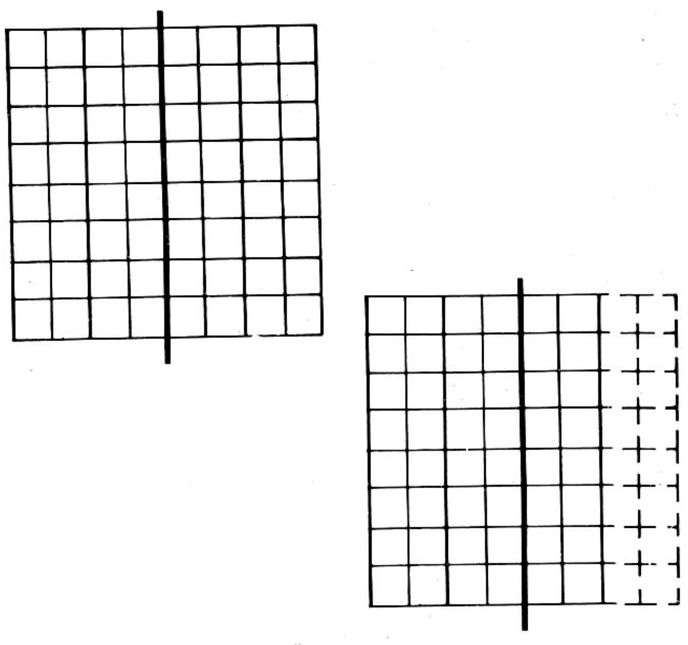
DEAR EDITOR

Could you please tell me how to get writing like in the Death Satellite program. I really enjoy using the Sega and think the magazine is great.

Martin Hazelwood

EDITORS REPLY

The Death satellite program redefines the computer character set using the PATTERN statement. The new characters are created in the same was as a Sprite (see issue 2) except that there is a 6 × 8 block instead of an 8 × 8 block.



On the left is a normal 8 × 8 sprite block with the dividing line marked. On the right is the 6 × 8 block used for redefining text characters. Notice where

THE LIGHT DAWNETH

At last the Government of the day has realised the importance of home computers and computers in general by reducing the sales tax from 40% to 10%.

While all of you owners were not able to take advantage of this sales tax reduction, which is unfortunate, it does mean that any future peripherals that you purchase will be far better priced than previously.

All of you who have written indicating interest in the monitor will be happy with the new price of $595.00. Unfortunately, it appears we will be unable to expect stocks before late January, as lead time on manufacturing overseas is approximately three months and Sega in Japan advise that they are having problems keeping pace with demand.

The following list gives you some idea of what our new retail prices on peripherals will be:

Product No	Description	Price
4202 4203	Printer) Adaptor for Printer)	$520.00
4208 4011	Data Recorder) Adaptor for Recorder)	89.95
4209	Disk Drive with I/o	995.00
4210	Colour Monitor	595.00

READER'S LETTERS

the dividing line is and that the last two columns on the right are ignored.

Text Character

e.g.

```
PATTERN C#65,"FF8181818181FF"
```

The C# stands for character and the number is the ASCII code number (see page 154 and its of the Sega level III manual) in this case "A" is redefined as a box, however, as only a 6 × 8 block is displayed the right hand side of the box is lost.

Try

```
PATTERN C#65,"FC8484848484FF"
```

Here is the program that creates the new characters in Death satellite program.

```
10 FORI=97TO122:READQ$:PATTERNC#I,Q$:NEXT
20 DATA 00F88888F88888,00F090F08888F8,00F880808080F8
30 DATA 00F0888888888F0,00F88080F080F8,00F88080F08080
40 DATA 00F88080B888F8,00888888F88888,00F8202020 20F8
50 DATA 00F810109090F0,0090A0C0A09088,00808080808080F0
60 DATA 00F8A8A8A8A888,0088C8A8988888,00F888888888F8
70 DATA 00F88888F88080,00708888A89068,00F08888F09088
80 DATA 00F880F80808F8,00F8202020202 0,00888888888888F8
90 DATA 00888888885020,008888A8A8A8F8,00885020508888
100 DATA 00885020202020,00F010204080F8
```

DEAR EDITOR

I have had my SC3ØØØ for a month now and I am still amazed at just what it can do. There is one thing I can not do, that is to draw a circle and write on top of it. I get a black box instead. Why is this?

Richard Roberts

EDITORS REPLY

On the Segas graphic screen once a dot is turned "on" it remains on until it is turned "off." If you try to turn "on" a dot twice you will obtain chunky graphics. To avoid this you must erase the dots in the area on which to print a message.

```
10 SCREEN 2,2:CLS
20 CIRCLE (100,100),50,3,1.25,,,BF
30 COLOR 1:CURSOR 85,97:PRINT "HELLO"
40 GOTO 40
```

Now the background colour shows through the hole in the circle. To fix this we must colour the background of that area in the colour of the circle. To do this add line 25.

```
25 COLOR ,3,(80,96)-(120,104)
```

PLEASE SEND YOUR QUERIES TO THE EDITOR:

GRANDSTAND LEISURE

P.O. BOX 2353
AUCKLAND

NEW VENTURE FOR GRANDSTAND

By Bill Fenton

Those of you who read the computer pages in the local press may have noticed that Granstand will be marketing an entirely new home computer cum small business machine which comes complete with 64K RAM built in cassette recorder and green screen monitor or colour monitor to be launched in New Zealand early next year.

It is the Amstrad CPC 464, a British computer which generated round-the-corner queues at a central London department store, early on the morning in June that it first went on sale.

The CPC 464 is designed by Amstrad to incorporate at the lowest possible price a comprehensive array of hardware in one convenient, self-contained package.

At this stage, all Sega owners will be wondering why on earth you have a Sega and not an Amstrad so let me reassure you. They are two entirely different machines with entirely different benefits, in fact the Sega SC3000 is the far easier to use and to learn programming.

Our commitment to Sega is complete and we will of course continue to support Sega with more software and peripherals as they come available.
We intend to combine the Sega and Amstrad Users' magazine which will make an extremely interesting publication and will give users of both Sega and Amstrad more opportunity to utilise international programs.

However, back to the Amstrad. In its basic configuration the CPC 464 includes a cassette recorder, aplha numeric style keyboard and green monitor, or colour monitor, and at its most sophisticated includes disk drive and dot matrix printer. It also boasts high resolution graphics, an 80 column text display, 64K of RAM and 32K of ROM, a fast, standard, 'basic', and a comprehensive user guide. For this reason it cannot be compared with existing home computers.

Earlier this year in June, British consumers queued up outside Rumbelow's, central London department store, waiting for the Amstrad to go on sale. Rumbelow's reported sales on that first day running at 20 an hour. A month later Britain's "Home Computer Weekly" reported Amstrad's Sales Director, Dickie Mould, as saying: "The initial interest was, perhaps, predictable.

But this level has been maintained and I am called every day by retailers who want to be supplied."

A comprehensive range of 90 software programs is also available under the 'Amsoft' name.

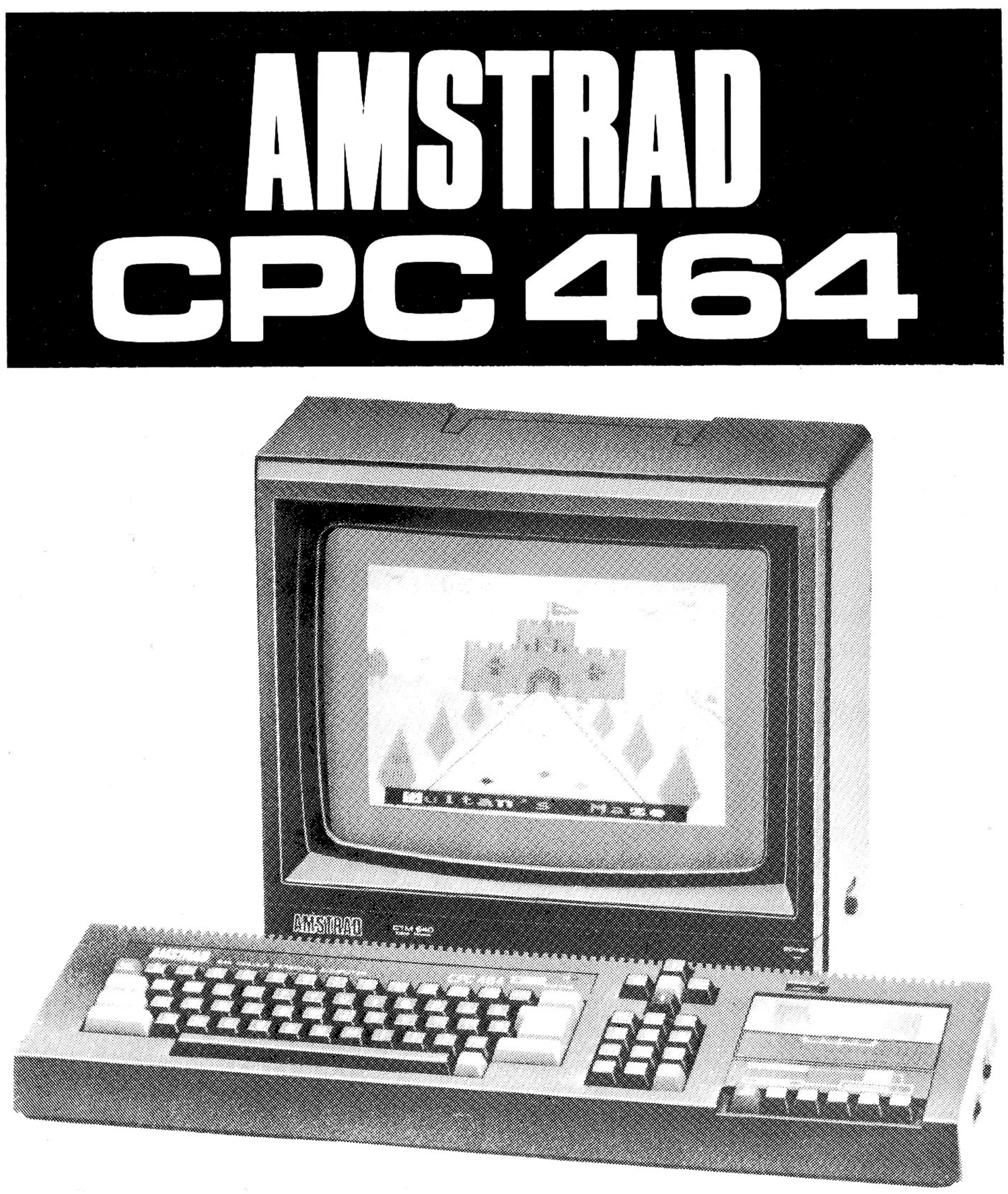
As a result of the number of British software houses lining up to write programs for Amstrad since the launch of the CPC 464, the number of Amsoft programs has increased from 90 to 120, and more are in prospect.

A major advantage in this for New Zealand is that the UK is fairly close to this country in its "way of doing things" and therefore, British software is inherently usuable here.

For Grandstand Leisure the Amstrad CPC 464 is further vindication of our campaign for more awareness among parents that: "a revolution is going on under their noses, irrevocably changing the world that their children have to contend with.

"The computer, particularly the home computer, is a two-edged sword. It can either widen the education gaps that already exist between children or, if we organise ourselves, it can close those gaps and at the same time push the process of learning up to a new plateau.

"When we first sounded that warning, six months ago, there were already two generations of home computer. Now there is a third; the technology is developing faster all the time and many parents are still ignoring the phenomenon, to their children's cost."



Program Dissection

Philip Bächler

PROGRAM DISSECTION 1

CLOCK

The following programs creates a clock using the Sega's built in timing function (TIME$). This clock can be set to any time and has an hourly chime.

By dissecting this program, you will learn how to use the TIME$ function and how to position lines and numbers in a circular fashion on the screen.

Line 1Ø:-

The REM statement tells the computer to ignore the rest of that line. REM is short for remark and is used for putting notes to yourself, in your program so that you can understand what that part of the program does.

Line 4Ø:-

This clears the text screen and prints the message inside quotation marks.

Line 5Ø:-

The computer prints the prompt to enter the time. Whatever information is entered is called T$. $ stands for string. The information may contain letters and punctuation. In this case we do not want letters, but the time.

e.g. 12:30:59

Would be 12 hours, 3Ø minutes, 59 seconds.

Line 6Ø:-

This tests to see if any information was entered in line 5Ø by seeing if T$ has got anything in it. If T$ is nothing (""), then the computer jumps to line 8Ø and the time is left set at what it was previously.

Line 7Ø:-

The computer will come to this line if Information has been entered on line 5Ø. Time$ (the computer's clock) will be reset to T$ (the time entered by the user). If the time is invalid, (contains letters etc), the program stops running.

Line 8Ø:-

This line sends the computer to graphics screen and clears this screen. PRINT CHR$(16) sets the computer to print in normal type. PRINT CHR$(17) would set the computer to print in double wide type. (Please see page 18 and 19 of Sega Level III Manual).

Line 9Ø:-

The POSITION command defines which point on the graphics screen is the origin (x = Ø, y = Ø). In this case the top left hand corner is the origin (this point is the origin normally) The two numbers that follow the coordinates in the brackets define the increased direction or the x and y axis (see page 110-113 of the Sega Level III Manual).

Line 1ØØ:-

This colours the entire background of the screen in white (code 15) and the border at the top and bottom of the screen white.

Line 11Ø:-

This gives the height to width ratio of the clock. Try changing this to Ø.75.

Line 12Ø:-

A large block of green is placed on the screen for the clock.

Line 13Ø:-

The coordinates 128,96 (the centre of the screen) is defined as the origin (point Ø,Ø). The x axis increases as it goes across the screen while the y axis increases UP the screen (it normally increases down the screen).

Line 14Ø:-

On the graphics screen a dot remains turned on until it is turned off. This means that if you draw say a box and then a circle on top ofthe box you get a "CHUNKY" circle.

This ERASES a circle from the filled box. The circle centre is point Ø,Ø (normally point 128,86 but it has been changed by line 13Ø) and a radius of 75 pixels the height to width rate is the variable RT (defined in line 11Ø).

Line 15Ø:-

Variable I is to be counted from 1 to 12 and is used to position the numerals on the clock face.

Line 16Ø:-

This line calculates the value of TH which is the angle at which the numbers are to be printed.

Line 17Ø:-

67 is multiplied with the SIN OF TH (which is the angle around the clock face) and with the circles height to width ratio. This gives the Y coordinate of where the numeral is to be printed.

Line 19Ø:-

Checks to see if I is greater than 1Ø. This is done as there are now two digits instead of one so the number has to be shifted to the left a little (X = X = 3). To give good result.

Line 2ØØ:-

The writing colour is set to number 2 which is light green (please see page 1ØØ of Sega III Manual).

Line 21Ø:-

This positions the cursor at position X − 6 along, Y down and prints the number 1.

Line 22Ø:-

This tells the computer to increase the value of X by 1 and if X is not greater than 12 then go back to line 16Ø and repeat the process. If I is greater than 12 (ie all the numerals are now open the clock face) the program continues to line 23Ø.

Line 23Ø:-

The computer defines L1 as 43 and L2 calculates LZ as 51.6 (this depends on the ratio). This line works out the length of the hour hand.

Line 24Ø:-

This calculates the length of the second hand.

Line 25Ø:-

This calculates the length of the hour hand.

Line 26Ø:-

This line of the program calculates the hours. The computer takes the first 2 characters of the left hand side of TIME$ and converts this smaller string into a number using the VAL function. The number is called H and contains the hours.

Line 27Ø:-

This calculates the minutes. The computer takes the middle 2 characters, 4 characters in from the left hand end of TIME$. This is converted to a number and stored in M (minutes).

Line 28Ø:–

The computer calculates HR which is the angle at which the hour hand is to be drawn.

Lines 29Ø-3ØØ:-

These calculate the X & Y coordinates for the end of the hour hand.

Line 31Ø:-

This line works out the angle around the clock face for the position of the minute hand.

Lines 32Ø-33Ø:-

These calculate the X & Y coordinates for the end of the hour hand.

Line 34Ø:-

The number of minutes (M) is now stored in M1. This is used to later on as a test to see if the minutes have changed.

Lines 35Ø-37Ø:-

Rem statements separating the intialising section of the program (lines 1Ø-34Ø) from the mainloop (lines 38Ø-67Ø).

Line 38Ø:-

This works out the number of seconds.

Line 39Ø:-

This tests to see if the time has changed by 1 second from t' 2 last time the second hand was moved. If the time has not changed the computer is sent back to line 38Ø. Line 39Ø stops the computer replotting the hands of the clock. This gives a more steady readable display.

Line 4ØØ:-

This time has now increased by 1 second as the computer has past line 39Ø. The computer updates T$ to equal TIME$ (TIME$ is the computer clock it increases itself automatically every second).

Line 41Ø-42Ø:-

These two lines perform the same function as lines 26Ø-27Ø. The computer sets the variables H & M (the hours and minutes).

Line 43Ø:-

This is defined as the new number of seconds and is used in line 39Ø for the test.

Line 44Ø:-

SR is the angle around the circle that the second hand is to be printed.

Line 45Ø-46Ø:-

These lines calculate the X & Y coordinates of the end of the second hand.

Line 47Ø:-

The computer checks to see if the number of minutes has increased. If they have not increased there is no point replotting or calculating the hour or minute hand so the computer is sent to line 57Ø. (see below).

Line 48Ø:-

This line is only executed if 1 minute has passed M(the new minutes) is not equal to M1, (the old minute value) MR is the angle at which the minute hand is plotted.

Line 49Ø-5ØØ:-

This calculates the X and Y coordinates of the end of the minute hand using the variable MR from the line above.

Line 51Ø:-

Sets M1 which used for testing in line 47Ø.

Lines 52Ø-57Ø:-

This erases the old hour, minute and second hands. This is done by removing a line from the old hour, minutes or seconds coordinate to the centre of the clock face (point Ø,Ø).

Line 58Ø-6ØØ:-

This routine places the new hands on the screen by drawing from the middle of the circle to the new X & Y coordinates.

Line 61Ø:-

This erases a block at the bottom of the screen where the time is printed.

Line 62Ø:-

This positions the cursor at the top left hand edge of the block erased above and sets the writing colour to black.

Line 63Ø:-

This prints the time in the box erased in line 61Ø.

Line 64Ø:-

The computer is sent to line 71Ø. It will work through the program there, until it sees the word RETURN. The program then jumps back to line 65Ø.

Line 71Ø:-

This is the second routine. This line tests to see if it is the 59th minute of any hour (if it is the computer continues to line 72Ø which tests to see if more than 57 seconds have passed in that minute. If 57 seconds have passed the computer continues to line 73Ø which is the hourly time signal).

Line 73Ø:-

This portion of the program creates the hour time signal. The computer plays an international A (see the section on sound in this issue). This is the pip before the hour.

Line 74Ø-75Ø:-

This creates a short delay by counting from 1 to 1Ø.

Line 76Ø:-

Sound Ø turns all the sound off.

Line 78Ø:-

This line test to see if the long tone of the hour should be played by testing MF. If MF = – 1 then the computer plays the tone.

Line 79Ø:-

The computer is sent here from line 71Ø to 72Ø. If it is not time for the hourly signal the computer, it produces a click. In line 79Ø the program tests to see if the volume of the hourly chime is greater than zero. This test is preformed to see if the chime can be played again more softly.

Line 81Ø:-

V is greater than zero. (see line 8ØØ), so a tone is played with a frequently of 88Ø hertz and a volume of V.

Line 82Ø:-

This decreases V (volume) by two. This makes the chime fade away.

Line 83Ø-84Ø:-

This checks to see if the minutes and seconds = Ø. If they do the hour has just been struck so the time signal may be played in line 85Ø.

Line 87Ø:-

This sends the computer back to line 65Ø. Every second the computer runs through the sound routine once it either produces a click in the 79Ø or a time signal in line 73Ø-76Ø and line 81Ø-86Ø.

Line 65Ø:-

The computer has now completed all moves and sounds required for the past second. It now makes the hour minute and second coordinates equal to the old hour, minute and second coordinates so that the old hands can be erased in the next second.

XH, YH Old hour coordinates
XM, YM Old minute coordinates
XS, YS Old second coordinates
HX, HY Old coordinates used for new time
MX, MY Minutes coordinates used for new time
SX, SY Second coordinates used for new time

Line 67Ø:-

The computer goes to line 37Ø to start the whole main loop again.

Line 28Ø:–

The computer calculates HR which is the angle at which the hour hand is to be drawn.

Lines 29Ø-3ØØ:-

These calculate the X & Y coordinates for the end of the hour hand.

Line 31Ø:-

This line works out the angle around the clock face for the position of the minute hand.

Lines 32Ø-33Ø:-

These calculate the X & Y coordinates for the end of the hour hand.

Line 34Ø:-

The number of minutes (M) is now stored in M1. This is used to later on as a test to see if the minutes have changed.

Lines 35Ø-37Ø:-

Rem statements separating the intialising section of the program (lines 1Ø-34Ø) from the mainloop (lines 38Ø-67Ø).

Line 38Ø:-

This works out the number of seconds.

Line 39Ø:-

This tests to see if the time has changed by 1 second from t' ₂ last time the second hand was moved. If the time has not changed the computer is sent back to line 38Ø. Line 39Ø stops the computer replotting the hands of the clock. This gives a more steady readable display.

Line 4ØØ:-

This time has now increased by 1 second as the computer has past line 39Ø. The computer updates T$ to equal TIME$ (TIME$ is the computer clock it increases itself automatically every second).

Line 41Ø-42Ø:-

These two lines perform the same function as lines 26Ø-27Ø. The computer sets the variables H & M (the hours and minutes).

Line 43Ø:-

This is defined as the new number of seconds and is used in line 39Ø for the test.

Line 44Ø:-

SR is the angle around the circle that the second hand is to be printed.

Line 45Ø-46Ø:-

These lines calculate the X & Y coordinates of the end of the second hand.

Line 47Ø:-

The computer checks to see if the number of minutes has increased. If they have not increased there is no point replotting or calculating the hour or minute hand so the computer is sent to line 57Ø. (see below).

Line 48Ø:-

This line is only executed if 1 minute has passed M(the new minutes) is not equal to M1, (the old minute value) MR is the angle at which the minute hand is plotted.

Line 49Ø-5ØØ:-

This calculates the X and Y coordinates of the end of the minute hand using the variable MR from the line above.

Line 51Ø:-

Sets M1 which used for testing in line 47Ø.

Lines 52Ø-57Ø:-

This erases the old hour, minute and second hands. This is done by removing a line from the old hour, minutes or seconds coordinate to the centre of the clock face (point Ø,Ø).

Line 58Ø-6ØØ:-

This routine places the new hands on the screen by drawing from the middle of the circle to the new X & Y coordinates.

Line 61Ø:-

This erases a block at the bottom of the screen where the time is printed.

Line 62Ø:-

This positions the cursor at the top left hand edge of the block erased above and sets the writing colour to black.

Line 63Ø:-

This prints the time in the box erased in line 61Ø.

Line 64Ø:-

The computer is sent to line 71Ø. It will work through the program there, until it sees the word RETURN. The program then jumps back to line 65Ø.

Line 71Ø:-

This is the second routine. This line tests to see if it is the 59th minute of any hour (if it is the computer continues to line 72Ø which tests to see if more than 57 seconds have passed in that minute. If 57 seconds have passed the computer continues to line 73Ø which is the hourly time signal).

Line 73Ø:-

This portion of the program creates the hour time signal. The computer plays an international A (see the section on sound in this issue). This is the pip before the hour.

Line 74Ø-75Ø:-

This creates a short delay by counting from 1 to 1Ø.

Line 76Ø:-

Sound Ø turns all the sound off.

Line 78Ø:-

This line test to see if the long tone of the hour should be played by testing MF. If MF = –1 then the computer plays the tone.

Line 79Ø:-

The computer is sent here from line 71Ø to 72Ø. If it is not time for the hourly signal the computer, it produces a click. In line 79Ø the program tests to see if the volume of the hourly chime is greater than zero. This test is preformed to see if the chime can be played again more softly.

Line 81Ø:-

V is greater than zero. (see line 8ØØ), so a tone is played with a frequently of 88Ø hertz and a volume of V.

Line 82Ø:-

This decreases V (volume) by two. This makes the chime fade away.

Line 83Ø-84Ø:-

This checks to see if the minutes and seconds = Ø. If they do the hour has just been struck so the time signal may be played in line 85Ø.

Line 87Ø:-

This sends the computer back to line 65Ø. Every second the computer runs through the sound routine once it either produces a click in the 79Ø or a time signal in line 73Ø-76Ø and line 81Ø-86Ø.

Line 65Ø:-

The computer has now completed all moves and sounds required for the past second. It now makes the hour minute and second coordinates equal to the old hour, minute and second coordinates so that the old hands can be erased in the next second.

XH, YH Old hour coordinates
XM, YM Old minute coordinates
XS, YS Old second coordinates
HX, HY Old coordinates used for new time
MX, MY Minutes coordinates used for new time
SX, SY Second coordinates used for new time

Line 67Ø:-

The computer goes to line 37Ø to start the whole main loop again.

```
10 REM ----------------------------
20 REM  Clock
30 REM ----------------------------
40 CLS:PRINT "TIME:                    HH:MM:SS"
50 INPUT "Enter the time... ";T$
60 IF T$="" THEN 80
70 TIME$=T$
80 SCREEN 2,2:CLS:PRINTCHR$(16);
90 POSITION (0,0),0,0
100 COLOR ,15,(0,0)-(255,191),15
110 RT=1.2
120 LINE (30,0)-(225,191),2,BF
130 POSITION (128,96),0,1
140 BCIRCLE(0,0),74,15,RT,,,BF
150 FOR I=1 TO 12
160 TH=RAD((3-I)*30)
170 Y=67*SIN(TH)*RT+4
180 X=67*COS(TH)-3
190 IF I>=10 THEN X=X-3
200 COLOR 2
210 CURSOR X-6,Y :PRINTI;
220 NEXT
230 L1=43:L2=RT*L1
240 L3=55:L4=RT*L3
250 L5=55:L6=RT*L5
260 H=VAL(LEFT$(TIME$,2))
270 M=VAL(MID$(TIME$,4,2))
280 HR=RAD(30*H+M/2)
290 HX=L1*SIN(HR)
300 HY=L2*COS(HR)
310 MR=RAD( 6*M)
320 MX=L3*SIN(MR)
330 MY=L4*COS(MR)
340 M1=M
350 REM
360 REM ----------------------------
370 REM
380 S=VAL(RIGHT$(TIME$,2))
390 IF S=S1 THEN 380
400 T$=TIME$
410 H=VAL(LEFT$(T$,2))
420 M=VAL(MID$(T$,4,2))
430 S1=S
440 SR=RAD( 6*S)
450 SX=L5*SIN(SR)
460 SY=L6*COS(SR)
470 IF M=M1 THEN 570
480 MR=RAD( 6*M)
490 MX=L3*SIN(MR)
500 MY=L4*COS(MR)
510 M1=M
520 HR=RAD(30*H + M/2)
530 HX=L1*SIN(HR)
540 HY=L2*COS(HR)
550 BLINE(0,0)-(XM,YM),1
560 BLINE(0,0)-(XH,YH),1
570 BLINE(0,0)-(XS,YS),6
580 LINE(0,0)-(HX,HY),1
590 LINE(0,0)-(MX,MY),1
600 LINE(0,0)-(SX,SY),6
610 BLINE(-24,-87)-(23,-95),15,BF
620 CURSOR -24,-88:COLOR 1
630 PRINT T$;
640 GOSUB 710
650 XH=HX:XM=MX:XS=SX
660 YH=HY:YM=MY:YS=SY
670 GOTO 370
680 REM
690 REM ----------------------------
700 REM
710 IF M<59 THEN 780
720 IF S<57 THEN 780
730 SOUND 1,440,15
740 FOR J=0 TO 10
750 NEXT
760 SOUND 0
770 MF=-1
780 IF MF=-1 THEN 830
790 IF V<8 THEN BEEP 1:BEEP 0
800 IF V<0 THEN 830
810 SOUND 1,880,V
820 V=V-2
830 IF M>0 THEN 870
840 IF S>0 THEN 870
850 SOUND 1,880,15
860 V=14:MF=0
870 RETURN
```


TURNERS
OF CHRISTCHURCH

Computer Specialists

Come and see Turners, the Sega specialists for the very latest in

SEGA SOFTWARE
PERIPHERALS

*

Expert advice from friendly staff

If you want to get the best for your Sega
see TURNERS first

Don't forget, order your Sega disk drive now from TURNERS

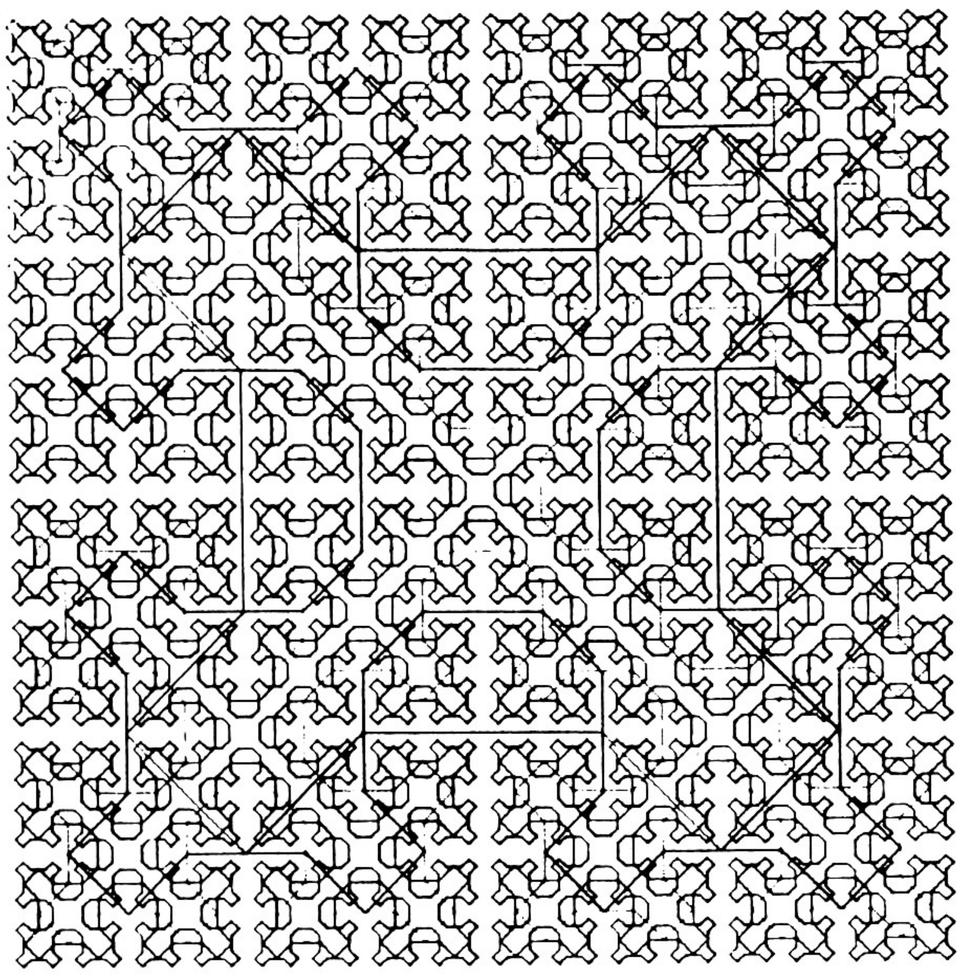
MAKE SURE YOU'RE NOT DISAPPOINTED

*

TURNERS

CNR COLOMBO AND TUAM STREETS
CHRISTCHURCH — PHONE 794-820

PRINTER DEMONSTRATION



```
100 REM *** PRINTER DEMO ***
120 LPRINT CHR$(18)
140 REM
150 H=384:X0=240:Y0=-X0:H=H/4
160 FOR I=1 TO 5
170 X0=X0-H:H=H/2
180 Y0=Y0+H
190 C=(I-1) MOD 4
200 LPRINT "C"+STR$(C)
205 X=X0:Y=Y0
210 LPRINT "M"+STR$(X0)+","+STR$(Y0)
220 GOSUB 1000:X=X+H:Y=Y-H:GOSUB 5000
222 GOSUB 2000:X=X-H:Y=Y-H:GOSUB 5000
224 GOSUB 3000:X=X-H:Y=Y+H:GOSUB 5000
226 GOSUB 4000:X=X+H:Y=Y+H:GOSUB 5000
230 NEXT I
290 REM
400 LPRINT "A"
420 END
```

```
999 REM
1000 IF I<=0 THEN 1070
1010 I=I-1
1020 GOSUB 1000:X=X+H:Y=Y-H:GOSUB 5000
1030 GOSUB 2000:X=X+2*H:GOSUB 5000
1040 GOSUB 4000:X=X+H:Y=Y+H:GOSUB 5000
1050 GOSUB 1000
1060 I=I+1
1070 RETURN
1080 REM
2000 IF I<=0 THEN 2070
2010 I=I-1
2020 GOSUB 2000:X=X-H:Y=Y-H:GOSUB 5000
2030 GOSUB 3000:Y=Y-2*H:GOSUB 5000
2040 GOSUB 1000:X=X+H:Y=Y-H:GOSUB 5000
2050 GOSUB 2000
2060 I=I+1
2070 RETURN
2080 REM
3000 IF I<=0 THEN 3070
3010 I=I-1
3020 GOSUB 3000:X=X-H:Y=Y+H:GOSUB 5000
3030 GOSUB 4000:X=X-2*H:GOSUB 5000
3040 GOSUB 2000:X=X-H:Y=Y-H:GOSUB 5000
3050 GOSUB 3000
3060 I=I+1
3070 RETURN
3080 REM
4000 IF I<=0 THEN 4070
4010 I=I-1
4020 GOSUB 4000:X=X+H:Y=Y+H:GOSUB 5000
4030 GOSUB 1000:Y=Y+2*H:GOSUB 5000
4040 GOSUB 3000:X=X-H:Y=Y+H:GOSUB 5000
4050 GOSUB 4000
4060 I=I+1
4070 RETURN
4080 REM
5000 LPRINT "D"+STR$(X)+","+STR$(Y)
5010 RETURN
```

PROGRAM OF THE MONTH

Each issue, we would like to invite Sega owners to send in their programs for evaluation and possible entry into the Magazine. Not only games programs, but utility programs, and machine code or basic routines. Each program will be judged on originality, usefullness, or play value, graphic content or design and layout, individually depending on its use or function.

Any programs which are considered to be commercially viable, separate contact will be made to agree terms. All other programs accepted for inclusion into the magazine, will receive a cassette program of the writers choice.

One program each issue will be singled out as being the ''Star Program of the Month'', entitling the writer to select any one of Sega's game cartridges as their prize.

THE FARMERS' TRADING COMPANY LIMITED

FARMERS' ASHBURTON & NORTH | FTC TIMARU & SOUTH

are pleased to

offer the ALL NEW

SEGA® SF 7000

SUPER CONTROL STATION

Featuring:

The latest standard 3'' compact floppy disk,
8 bit parallel port (centronics)
R.S. 232C Port (Baud rate: 300-9600)
Main memory 64K bytes RAM
Main language disk basic (supplied)

$995

From $99.50 deposit
From $11.10 weekly

Limited stocks available from the following FTC FARMERS'

Bank St Whangarei, Colombo St Christchurch, Cuba St Wellington, Devon Mall Tauranga, Pakuranga Mall Auckland, Hobson St Auckland, Manukau Mall Manukau City, Shore City Auckland, Browns Bay Auckland, George St Dunedin, Collingwood St Hamilton.

Call into the above stores or order by mail

Mail order coupon To F T C the Farmers Trading Co Private Bag Auckland.

Name .....

Address .....

Phone .....

If paying cash please enclose cheque.
If buying on an F T C account please enclose 10% deposit ($99.50) and quote the account number

Signature ..... We guarantee satisfaction

BEGINNERS PROGRAMMING

Now that we have covered some simple text and graphics commands, we can now start to create some more interesting programs using information accepted from the keyboard or joysticks. We have already seen in issue one, how to enter words (strings) and numbers (variables) into the computer using the statement INPUT.

e.g. 1

```
10 INPUT A$
20 PRINT A$;
30 GOTO 20
```

e.g. 2

```
10 INPUT "Enter a number please...";X
20 FOR I=1 TO X
30 PRINT I;
40 NEXT I
```

INKEY $

This is another way to get information from the keyboard. The computer scans the keyboard to see which, if any, keys are being pressed.

```
10 PRINT INKEY$
20 GOTO 10
```

The computer looks at the keyboard and prints whatever key is being pressed. Blank lines are printed when no keys are pressed. INKEY$ is used a lot in games as the [CR] key does not need to be pressed for information to be entered. It is also possible to see if individual keys have been pressed. Unlike INPUT, INKEY$ will scan the keyboard while the computer is using both the graphics and text screen.

e.g. 1

```
10 SCREEN 2,2:CLS
20 PRINT "PRESS [C] TO CONTINUE"
30 A$=INKEY$
40 IF A$="C" THEN 60
50 GOTO 30
60 CLS:PRINT "YOU HAVE CONTINUED"
70 GOTO 70
```

e.g. 2

```
10 SCREEN 1,1:CLS
20 X$="---":Y=10
30 A$=INKEY$
40 IF A$="U" THEN 100
50 IF A$="D" THEN 120
60 GOTO 30
100 CLS:CURSOR 10,Y:PRINT X$
110 Y=Y-1:GOTO 30
120 CLS:CURSOR 10,Y:PRINT X$
130 Y=Y+1:GOTO 30
```

Example 1

This program operates on the graphic screen (screen 2,2), A$ records the last key pressed. When the C key is pushed the program jumps to line 60 and the message is displayed.

Example 2

This operates on text screen. The keyboard is scanned for the U (up) and D (down) keys. If one of these keys are pressed, the computer moves the cursor up or down and prints X$ (X$ is minus signs). If you hold down either key too long, the program will stop working as you are trying to print information at a position that does not exist (off the top or bottom of the screen). To stop the program doing this we must test to see if Y (the line on screen on which the X$ is to be printed) is between 0 and 23 (there are 24 lines on the screen numbered from 0-23). To do this make the following alterations:

```
40 IF A$="U" THEN 95
50 IF A$="D" THEN 115

95 IF Y=0 THEN Y=1

115 IF Y=23 THEN Y=22
PRINT CHR$(249)
```

Each letter or key has its own number. This is its ASCII code number. ASCII the American Standard Code for Information Interchange, is the code structure used internally in most personal computers to represent letters, numbers and special characters. (See page 150-155 of level III manual). This allows us to print or test for any character. For example printing the character with ASCII code 249 produces a face.

The CHR$ function prints the character with the same ASCII code enclosed in the brackets.

E.g.
Print CHR$ (65) Produces A
Print CHR$ (66) Produces B
Print CHR$ (97) Produces a
Print CHR$ (98) Produces b

The CHR$ function can also be used to test for characters by checking the ASCII codes.

To scan the keyboard for the arrow keys, we test for the control characters for the four directions. (Please see Pages 18 and 19 of the Level III Sega Manual).

CHR$ (28) → Key
CHR$ (29) ← Key
CHR$ (30) ↑ Key
CHR$ (31) ↓ Key

The following program works on the graphic screen and tests to see if anyone of the arrow keys have been pressed. If one of the arrows has been pressed, the computer moves the cursor one pixel or dot in that direction and places a black dot there, thus leaving a trail of dots.

```
10 SCREEN 2,2:CLS
20 X=127:Y=95
30 A$=INKEY$
40 IF A$=CHR$(28) THEN X=X+1:GOTO 100
50 IF A$=CHR$(29) THEN X=X-1:GOTO 100
60 IF A$=CHR$(30) THEN Y=Y-1:GOTO 100
70 IF A$=CHR$(31) THEN Y=Y+1:GOTO 100
80 GOTO 30
100 IF X<0 THEN X=0
110 IF X>255 THEN X=255
120 IF Y<0 THEN Y=0
130 IF Y>191 THEN Y=191
140 PSET(X,Y),1
150 GOTO 30
```

Line 10:-

The computer switches to the graphics screen and clears the screen of any drawings or words already there from the last program.

Line 20:-

This defines the coordinates where the computer will start drawing from. The X coordinate is the number of dots across the screen from the top left hand corner, while the Y coordinate is the number of dots down the screen. Position (127,95) is about the middle of the screen.

Line 30:-

The keyboard is scanned and any key that is pressed is recorded in A$.

Line 40:-

This tests to see if the → arrow key (CHR$ (28)) has been pressed. If it has been X is increased by 1 (so the line moves to the right). The program then jumps to line 100. If the → arrow has not been pressed the computer continues onto line 50.

Line 50:-

If the ← arrow key (CHR$ (29)) has been pressed the X variable is decreased by 1 (the line moves to the left). The program then jumps to line 100. If the ← arrow has not been pressed the computer does not change X and continues to line 60.

Line 6Ø:-

This tests to see if the ↑ key has been pressed. If this key (CHR$ (3Ø)) has not been pressed the computer jumps to line 7Ø. If the ↑ key has been pressed the Y variable is decreased and the computer jumps to line 1ØØ.

Line 7Ø:-

If the ↓ arrow key has been pressed the computer increases Y and jumps to line 1ØØ.

Line 8Ø:-

The computer only reaches this line if none of the arrow keys have been pressed. The program now jumps back to line 3Ø and starts testing all over again.

Line 1ØØ

This test to see if X less than Ø. If X is less than Ø (in other words, off the screen), x is set to Ø (back on the screen). This stops the program crashing or giving errors.

Line 11Ø:-

This tests to see if the dot is going off the right hand side of the screen. If it is (X is greater than 255), it is brought back on the screen (X is set to 255).

Line 12Ø:-

The computer checks to see if Y is less than zero (if Y is less than zero it is off the top of the screen, causing the program to "crash").

Line 13Ø:-

This tests to see if the dot is coming off the bottom of the screen. If it is the computer stops it by making Y equal 191 (the bottom line of the screen).

Line 14Ø:-

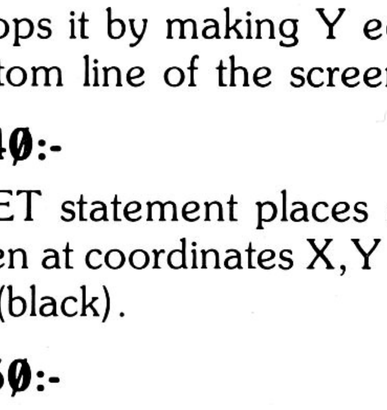
The PSET statement places a dot on the screen at coordinates X,Y in colour code 1 (black).

Line 15Ø:-

The computer jumps back to line 3Ø and starts scanning all over again.

STICK

This statement tells the computer to scan the joystick to see in what direction it is pointing. The keyboard is ignored. The computer returns a number between Ø and 8 depending on the direction in which the joystick is pushed.



Lines 1-4:-

```
1 REM THIS PROGRAM PRINTS THE
2 REM DIRECTION IN WHICH THE
3 REM JOYSTICK IS PRESSED. THIS IS
4 REM A NUMBER FROM 0-8.
10 S=STICK(1)
20 PRINT S;
30 GOTO 10
```

Rem statements. These are notes to the user. The computer ignores them.

Line 1Ø:-

The first joystick is scanned and the value of the direction that it is pushed is stored in S.

Line 2Ø:-

Displays S (the direction in which the joystick is pressed).

Line 3Ø:-

Jumps back to line 1Ø thus repeat the scan.

It is possible to scan both joysticks individually.

The statement stick (1) scans the player 1 joystick
The statement stick (2) scans the player 2 joystick

The above demonstration program that draws on the screen can be rewritten to work with the joystick by replacing lines 3Ø-7Ø with the following:

```
30 A=STICK(1)
40 IF A=3 THEN X=X+1:GOTO 100
50 IF A=7 THEN X=X-1:GOTO 100
60 IF A=1 THEN Y=Y-1:GOTO 100
70 IF A=5 THEN Y=Y+1:GOTO 100
```

ON GOTO

In the program above, lines 4Ø-8Ø can be replaced with the following lines.

This increases the speed of the program as it does not have to read line 4Ø-8Ø every time it runs, as it jumps directly to the line required.

```
10 SCREEN 2,2:CLS
20 X=127:Y=95
30 A=STICK(1)
40 ON A GOTO 50,30,60,30,70,30,80
45 GOTO 30
50 Y=Y-1:GOTO 100
60 X=X+1:GOTO 100
70 Y=Y+1:GOTO 100
80 X=X-1:GOTO 100
100 IF X<0 THEN X=0
110 IF X>255 THEN X=255
120 IF Y<0 THEN Y=0
130 IF Y>191 THEN Y=191
140 PSET(X,Y),1
150 GOTO 30
```

Line 4Ø tells the computer to goto a line depending on the value of the variable A.

This table shows which line number the computer would goto in the above case for various values of a computer.

Value of A	Line Number sent to
Ø	45
1	5Ø
2	3Ø
3	6Ø
4	3Ø
5	7Ø
6	3Ø
7	8Ø
8	3Ø
1Ø	45

For example if A is 3 (the joystick is in the right hand direction) the computer jumps to line 6Ø.

On Gosub

This is very similar to the On Goto statement and has the same format.

```
5 A=INT(RND(1)*6+1)
10 ON A GOSUB 50,60,70,80,90,100
20 GOTO 5
50 PRINT "1ST PLACE";A:GOTO 200
60 PRINT "2ND PLACE";A:GOTO 200
70 PRINT "3RD PLACE";A:GOTO 200
80 PRINT "4TH PLACE";A:GOTO 200
90 PRINT "5TH PLACE";A:GOTO 200
100 PRINT "6TH PLACE";A:GOTO 200
200 FOR DE=1 TO 100:NEXT DE
210 RETURN
```

n line 1Ø the computer creates a random number between 1 and 6, depending on the value of the random number, the computer goes to one of the line numbers listed after the word Gosub.

For example, if the random number is 4, the computer goes to line 8Ø.

When the computer sees the word RETURN it jumps back to the statement just after the last GOSUB statement that was performed. In this case the program is sent from line 21Ø to line 2Ø.

Strig

The Strig functions scans the fire buttons on the joystick. The computer returns a value between Ø-3.

Value Returned	Function
Ø	Neither buttons pressed
1	Left button pressed
2	Right button pressed
3	Both left & right buttons pressed

The strig function can scan either joystick

STRIG (1) : Joystick 1
STRIG (2) : Joystick 2

In the program lines 1Ø and 2Ø intialise the graphics screen, define X and Y and sets (the colour to 1 (Black).

```
10 SCREEN 2,2:CLS
20 X=127:Y=95:C=1
30 A=STICK(1)
40 ON A+1 GOTO 200,50,30,60,30,70,30,80,30
50 Y=Y-1:GOTO 100
60 X=X+1:GOTO 100
70 Y=Y+1:GOTO 100
80 X=X-1:GOTO 100
100 IF X<0 THEN X=0:GOTO 120
110 IF X>255 THEN X=255
120 IF Y<0 THEN Y=0:GOTO 140
130 IF Y>191 THEN Y=191
140 PSET(X,Y),C
200 B=STRIG(1)
210 IF B>0 THEN C=RND(1)*14+1:BEEP
220 GOTO 30
```

Line 40:-

Acts on the information from the joystick. If A is 0 (the joystick is in the REST position) the computer jumps to line 200 where the routine to scan the fire buttons is located.

Line 200:-

This scans the player 1 joystick fire buttons. A value between 0 and 3 is returned depending on which, if any, of the fire buttons are pressed. This value is stored in B.

Line 210:-

If either or both fire buttons have been pressed a value of 1-3 will be stored in B. This line tests to see if a button has been pushed by seeing if B is greater than zero (if B is zero the fire buttons have not been pushed).

If B is greater than 0 the colour is gauged to a random number between 1-14 and the computer produces a beep.

Time $

The Sega has its own input clock. This calculates how much time has elapsed since the computer was turned on or last reset.

Print Time $

The computer will display eight characters in the following format:

"Hours:Minutes:Seconds"

You can reset the value of time $ to any time you wish.

```
10 INPUT "Enter the current time...";T$
20 CLS
30 TIME$=T$
40 CURSOR 15,10:PRINT TIME$
50 GOTO 40
```

The above program prompts you to enter the time and the computer automatically updates this time using its TIME$ function. We do not see the updating as the computer is printing the time repeatedly in one place. (position 15,10).

```
10 SCREEN 2,2:CLS
20 CURSOR 50,180:PRINT "PRESS ANY KEY TO CONTINUE":TIME$="00:00:00"
30 A$=INKEY$
40 IF A$<>"" OR TIME$>"00:00:10" THEN 60
50 GOTO 30
60 CLS
70 COLOR 6:PRINT "YOU HAVE CONTINUED"
80 GOTO 80
```

This program ask you to press to continue. If you do not do so in 10 seconds the program moves on automatically to line 60. TIME$ has many functions in games and educational programs to record the amount of time it takes you to perform a certain task.

ADDENDUM

We wish to inform you of alterations which can be made to some of the earlier copies of the Bugaloo cassette program which will ensure the program will run correctly.

```
50 GOTO30:DATA7,1,6,1,4,1,3,1,1,2,0,3,1,3,3,3,4,3,4,4,2,5,0,5,12,1,11,1
,10,1,9,1,8,2,7,3,6,4,6,5,7,5,8,5,9,5,9,3,10,3,11,3,18,1,17,1,16,1,15,1
,14,2,14,3,13,4,13,5,14,5,16,5,18,5,18,4,18,3,17,3,23,1,22,1,21,2,21,3,
21,4,21,5,22,3,23,3,25,2,25,3,25,4,25,5
51 DATA0,0
```

It has been brought to our attention that the Cube-It cassette program ceases functioning when the snake catches the player on the bottom cube. To rectify this we suggest that you make the following alleration to the program.

```
512 FORI=YTO25STEP-4:SPRITE0,(X,I),47:SOUND1,530-I*3,8:SPRITE0,(X,I-2),51:SOUND2,542-I*3,8:NEXT:SPRITE0,,,0:GOTO640
```

You may have also noticed the Chess game does not allow people to play with joysticks. If you wish to have the program working correctly please change Line 18:

```
18 CLS:MO=1
```

FIREMAN BY SEGA

The idea of this basic game is to rescue people from a burning building and put out the fire. However, you must be very careful when scaling the side of the building not to get burnt to a crisp. You are armed with a portable fire extinguisher that can put out up to five burning windows. Once this is empty you can refill it by returning to the red fire engine. To move onto the next stage of your task you must rescue at least 12 people and bring them back down to the ground. As you are not Superman you can only carry three people at a time. Should you attempt to carry more than three people you all plunge to your deaths on the footpath. The game gets progressively harder and harder as the fire spreads rapidly through out the building and certain windows become too hot and melt. These windows become deadly when they are a red blob and can appear any where in the building.

INSTRUCTIONS

Use the four cursor arrow keys to control the fire men and the space bar to squirt the fire extinguisher.

We suggest that you use the filled block graphic character located on the V key in place of the small E in lines 52Ø-58Ø as this gives a more readable display.

```
1 REM ************
2 REM * FIRE.MAN *
3 REM ************
10 PATTERN C#79,"7088888888887000"
20 PATTERN C#48,"708898A8C8887000"
30 PATTERN C#55,"F8085020504040Ø0"
40 MAG0:GOSUB410:GOSUB500
50 HC=0
60 SC=0:MA=3:ST=1
70 MX=152:MY=168:WA=20:WWA=5:CC=0:CH=0:CTT=10+ST:C=4:FC=0:IF CTT>25THEN CTT=25
80 SCREEN2,2:CLS:COLOR,14,(0,0)-(255,191),6:COLOR,3,(0,176)-(255,191):COLOR,15,(
205,0)-(255,175):GOSUB800
90 FX=INT(RND(1)*17+4)*8:FY=INT(RND(1)*17+4)*8
100 REM  ** MAIN LOOP **
110 HO=INT(RND(1)*4)
120 IF HO=0 THEN FX=FX+8
130 IF HO=1 THEN FX=FX-8
140 IF HO=2 THEN FY=FY+8
150 IF HO=3 THEN FY=FY-8
160 IF FX<32 THEN FX=32
170 IF FX>160 THEN FX=160
180 IF FY<32 THEN FY=32
190 IF FY>160 THEN FY=160
200 BLINE(FX,FY)-(FX+7,FY+7),15,BF
210 CURSOR FX,FY:COLOR6:PRINTCHR$(135);
220 R=INT(RND(1)*10)
225 IF R<3THENBX=INT(RND(1)*17+4)*8:BY=INT(RND(1)*17+4)*8:BLINE(BX,BY)-(BX+7,BY+
7),15,BF:CURSOR BX,BY:COLOR6:PRINTCHR$(136);
230 K$=INKEY$
240 IF K$=CHR$(28)THEN IF MX<153THEN MX=MX+8
250 IF K$=CHR$(29)THEN IF MX>39THEN MX=MX-8
260 IF K$=CHR$(30)THEN IF MY>39THEN MY=MY-8
270 IF K$=CHR$(31)THEN IF MY<161THEN MY=MY+8
280 IF K$=" "THEN IF WWA>0THEN GOSUB1100
290 B=MY+2:A=VPEEK(INT(B/8)*256+INT(MX/8)*8+B MOD8)
300 SPRITE0,(MX,MY),0,C:BEEP
310 IF A=156OR A=24THEN GOSUB1190
320 IF A=90THEN CC=CC+1:BLINE(MX,MY)-(MX+7,MY+7),15,BF:SC=SC+200:GOSUB1300:IF CC
>2THEN C=6
330 IF MY=168AND CC=3THEN SC=SC+1000:C=4:GOSUB1300:GOSUB1400
340 IF MY=168AND CC>0 THEN GOSUB1400
350 IF MX=48AND MY=168AND WWA<1THEN GOSUB1500
360 IF CC>3THEN GOSUB1190
370 IF MA<1OR FC>200THEN GOTO1700
380 IF CH>CTT THEN GOTO1600
390 FC=FC+1:GOTO110
400 REM  ** CHARACTER DEFINITION **
410 FORI=135TO143:READ A$:PATTERNC#I,A$:NEXT I
420 FORI=0TO3:READ A$:PATTERNS#I,A$:NEXT I
430 DATA 298C9C366343763C,C324182442FF5A81,19803A7C7EBD1046
440 DATA 00425A5A3C3CFF00,007E424242427E00,000000183F274444
450 DATA 030F3CF0C000067F,FEFFABFF6C93EF38,46E2C27F6DF3120C
460 DATA 2A3ABAFE38187EC3,FFE7C3C3E7FFEF38
470 DATA 8090F8F8FEFEEA38,A1B9B97E381899FF
480 RETURN
500 REM  ** FIRST SCREEN **
510 SCREEN2,2:CLS:COLOR,15,(0,0)-(255,191),15:COLOR8
```

```
520 CURSOR6,0:PRINT "  eeee eee eeee  eeeee e   e   e   e   e";
530 CURSOR6,6:PRINT "  e     e  e   e e     ee ee  e e  e   e";
540 CURSOR6,12:PRINT"  e     e  e   e e     e e e e   e ee  e";
550 CURSOR6,18:PRINT"  eeee  e  eeee  eeeee e   e e   e e e e";
560 CURSOR6,24:PRINT"  e     e  e e   e     e   e eeeee e  ee";
570 CURSOR6,32:PRINT"  e     e  e  e  e     e   e e   e e   e";
580 CURSOR6,40:PRINT"  e    eee e   e eeeee e   e e   e e   e";
590 CURSOR100,60:COLOR4:PRINT"GRANDSTAND"
600 COLOR,11,(0,70)-(255,191)
610 CURSOR40,80:COLOR1:PRINT"** CHARACTER TABLE **"
620 SPRITE0,(40,96),0,4:CURSOR48,96:COLOR6:PRINT".....FIRE MAN [CURSOR KEY]"
630 CURSOR40,105:COLOR5:PRINTCHR$(137):CURSOR48,105:PRINT".....WATER [SPACE BAR]"
640 CURSOR40,114:COLOR6:PRINTCHR$(135):CURSOR48,114:PRINT".........50 points"
650 CURSOR40,123:COLOR1:PRINTCHR$(138):CURSOR48,123:PRINT".........200 points"
660 CURSOR40,132:COLOR6:PRINTCHR$(136):CURSOR48,132:PRINT".........100~1000 poin
ts"
670 CURSOR80,142:COLOR6:PRINTCHR$(140);CHR$(141):CURSOR80,150:PRINTCHR$(142);CHR
$(143)
680 FORI=0TO248 STEP8
690 SPRITE1,(I,150),1,15:SPRITE2,(I+8,150),2,15
700 SOUND1,587,12:FORT=1TO30:NEXT T:SOUND1,784,7:FORT=1TO20:NEXT T,I:SOUND0
710 CURSOR64,160:COLOR1:PRINT"Push [S] Key To Start"
720 IF INKEY$="S"THEN RETURN
730 GOTO720
800 REM  ** GAME SCREEN **
810 LINE(30,30)-(170,170),13,B
820 FOR I=32TO160 STEP8:FOR L=32TO164 STEP8:CURSOR L,I:PRINTCHR$(139):NEXT L,I
830 LINE(48,169)-(168,176),13,B
840 CURSOR32,160:COLOR6:PRINTCHR$(140);CHR$(141):CURSOR32,168:PRINTCHR$(142);CHR
$(142)
850 CURSOR48,8:COLOR6:PRINTCHR$(135);CHR$(135);CHR$(135);" FIRE MAN ";CHR$(135);
CHR$(135);CHR$(135)
860 CURSOR200,8:COLOR6:PRINT"HI SCORE"
870 CURSOR200,32:COLOR4:PRINT"SCORE":CURSOR200,56:PRINT"FIRE MAN"
880 CURSOR200,80:PRINT"MEN":CURSOR200,104:PRINT"WATER"
890 CURSOR200,128:COLOR2:PRINT"STAGE"
900 FORI=0TO184 STEP8
910 SPRITE1,(I,168),1,15:SPRITE2,(I+8,168),2,15
920 SOUND1,587,12:FORT=1TO30:NEXT T:SOUND1,784,7:FORT=1TO20:NEXT T,I
930 SOUND0:FORI=0TO30
940 CX=INT(RND(1)*17+4)*8:CY=INT(RND(1)*17+4)*8
950 BLINE(CX,CY)-(CX+7,CY+7),15,BF
960 CURSORCX,CY:COLOR1:PRINTCHR$(138):NEXT I
970 CURSOR200,16:COLOR1:PRINT HC
980 CURSOR200,136:PRINT ST:COLOR1:CURSOR200,40:PRINT SC
990 CURSOR200,64:PRINT MA:CURSOR200,88:PRINT CH:CURSOR200,112:PRINT WA
1000 RETURN
1100 REM  ** WATER **
1110 B=MY-6:A=VPEEK(INT(B/8)*256+INT(MX/8)*8+BMOD8)
1120 IF A=156THEN BEEP:SC=SC+50
1130 IF A=24THEN BEEP:SC=SC+INT(RND(1)*10+1)*10
1140 IF A=90THEN BEEP:BEEP:BEEP:SC=SC-100
1150 GOSUB1300
1160 BLINE(MX,MY-8)-(MX+7,MY+1),15,BF
1170 CURSOR MX,MY-8:COLOR7:PRINTCHR$(137);
1180 WWA=WWA-1:RETURN
1190 REM  ** FIRE MAN DEAD **
1200 CC=0:FORT=0TO100:BEEP1:BEEP0:NEXT T:FOR I=MY TO176 STEP8:SPRITE0,(MX,I),3,6
:NEXT I:C=4:MA=MA-1:MX=152:MY=168
1210 BLINE(200,64)-(231,71),15,BF
1220 CURSOR200,64:COLOR4:PRINT MA:RETURN
1300 REM  ** SCORE **
1310 BLINE(200,40)-(255,47),15,BF
1320 CURSOR200,40:COLOR1:PRINT SC:RETURN
1400 REM  ** MEN **
1410 CH=CH+CC:CC=0:BLINE(200,88)-(231,95),15,BF
1420 CURSOR200,88:COLOR4:PRINT CH:RETURN
1500 REM  ** WATER **
1510 IF WA=0THEN RETURN
1520 SOUND 1,262,15:FORT=1TO10:NEXT T:SOUND0:WA=WA-5:WWA=5:BLINE(200,112)-(231,1
19),15,BF
```

```
1530 CURSOR200,112:COLOR4:PRINT WA:RETURN
1600 REM  ** STAGE CLEARED **
1610 BLINE(48,64)-(136,71),15,BF
1620 CURSOR60,64:COLOR4:PRINT"NICE RESCUE!":ST=ST+1
1630 FORI=1TO30:BEEP:NEXT I:GOTO70
1700 REM  ** GAME OVER **
1710 FORI=1TO30:BEEP:NEXT I:IF SC>HC THEN HC=SC
1720 SCREEN2,2:CLS:COLOR,6,(0,0)-(255,191),6:COLOR,11,(0,40)-(255,120):CURSOR64,
40:PRINT"GAME OVER"
1730 CURSOR64,56:COLOR6:PRINT"HI-SCORE:";:COLOR15:PRINT HC
1740 CURSOR64,72:COLOR4:PRINT"YOUR SCORE:";:COLOR15:PRINT SC
1750 CURSOR48,96:COLOR4:PRINT"TRY AGAIN? PUSH [R] KEY"
1760 CURSOR48,112:COLOR6:PRINT"END? PUSH [E] KEY"
1770 A$=INKEY$
1780 IF A$="R"THEN GOTO60
1790 IF A$="E"THEN END
1800 GOTO1770
```


SPIDERSOFT'S CHRISTMAS SELECTION for the SEGA

1. SPIDERGRAPHICS by Grant Anderson

* SPIDERGRAPHICS is a LOGO program which allows common LOGO commands including REPEATS and PROCEDURE DEFINITION. TEXT PRESENTATION is possible on the graphics screen and commands such as CIRCLE, SQUARE and PAINT take advantage of SEGA'S capabilities.

* A ''DRAW'' mode allows key controlled finishing of pictures. Procedure can be saved with the program so you can make your own commands.

* A full set of instructions and demonstration procedures accompany each program.

* PARTICULARLY SUITABLE FOR INTERMEDIATE AND SECONDARY PUPILS. **PRICE $19.95**

THE SPIDERSOFT EARLY LEARNER SERIES

This series of six programmes is produced by a N.Z. teacher for the pre-school to early primary area. Entertaining games encourage children to practise important skills.

Titles	Age Range	Titles	Age Range
1 Comparing and arranging shapes	3 to 5	2 Numeral recognition	4 to 5
3 Addition + subtraction level 1	5 to 7	4 Addition + subtraction level 2	6 to 8
5 Adding suffixes (to words)	5 to 8	6 Alphabet + keyboard skills	5 to 8

Price: $18.95 each or any 2 programs for $29.99
Special offer: The complete series (6 programmes) for $64.95.

SPIDERSOFT ''TIMES TABLES''

It tests, it insists on corrections, then it rewards with the exciting maze game ''Stealth.'' Persistence brings increasing rewards.

For age 8 upwards.

ORDER FORM

RETAILER ENQUIRIES ARE WELCOME.

To SUPERCOM DATA SERVICES LTD., Box 34-313, Birkenhead, Auckland.

Name .....

Address .....

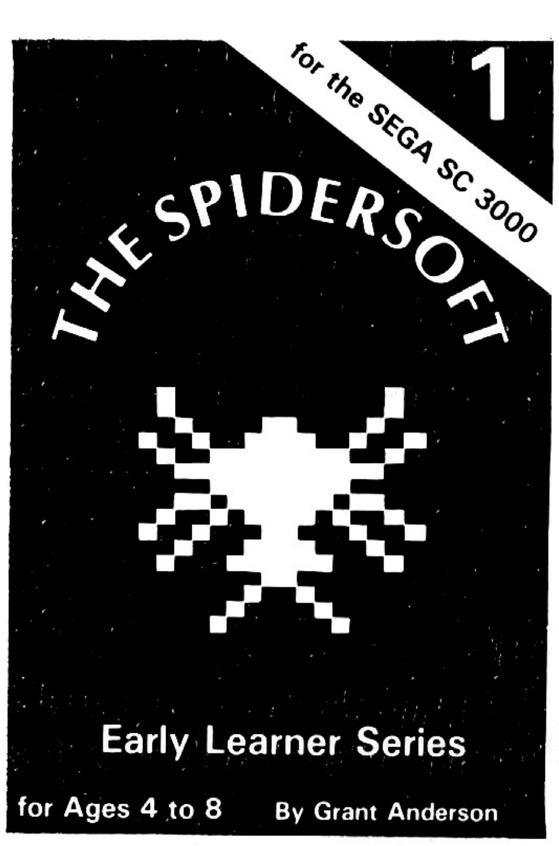
Payment by Cheque ☐ Bankcard ☐ Visa ☐

Card No. .....

Expiry Date ..... Signature .....

Tick the titles you wish to order

SPIDERGRAPHICS	☐			**.....$19.95**
EARLY LEARNER SERIES	**1** ☐	**4** ☐		**.....$18.95 ea**
	2 ☐	**5** ☐		**.....$29.99 for 2**
	3 ☐	**6** ☐		**.....$64.95 for 6**
TIMES TABLES	☐			**.....$19.95**
TOTAL COST				



CONVERSION CHART

This chart shows you how to convert commands in the program listings for other computer into commands which will work on your computer. To use this chart look along the top line until you find the name of the computer whose program you want to convert to the Sega. Then look down the column until you come across the command that you wish to convert. The Sega equivalent is on the far left-hand side.

SEGA SC3000	Description of command	BBC	SPECTRUM	DRAGON	APPLE
SCREEN 2.2	Set the graphics mode. (You need a mode with at least four colours.	MODE 1	REM (Computer automatically draws in high resolution graphics. See footnote 1.)	PMODE 3:SCREEN 1,Ø	HGR
SCREEN 1,1:CLS	Clear the text screen.	CLS	CLS	CLS	HOME
SCREEN 2,2:CLS	Clear the graphics screen.	CLG	CLS	PCLS	HGR
COLOR N	Set the colour for drawing lines.	GCOLØ,N	INK N	COLOR N,1	HCOLOR=N
CURSOR X,Y	Move the cursor to point X,Y	MOVE X,Y	PLOT OVER 1; INVERSE 1;X,Y	LINE(X,Y) – (X,Y),PRESE-T	HPLOTX,Y
LINE – (X,Y)	Draw a line from last cursor position to new position X,Y	DRAWX,Y	DRAWX,Y	LINE – (X,Y),PSET	HPLOTTOX,Y
PSET(X,Y)	Plot a point at position X,Y	PLOT69,X,Y	PLOTX,Y	LINE(X,Y) – (X,Y),PSET	HPLOTX,Y
PRINT TAB (X,Y)	Print text or variable at position X,Y	PRINT TAB(X,Y)	PRINT AT Y,X (See footnote 2.)	PRINT @32*Y + X,"———" (See footnote 2.)	VTAB(Y + 1);HTAB(X + 1);PRINT"———" (See footnote 2.)
BEEP	Make a short beep noise.	VDU7	BEEP1,1	SOUND9Ø,1	PRINTCHR$(7);
X$=INKEY$	Test to see if a key on the keyboard has been pressed.	X$=INKEY$(Ø)	X$=INKEY$	X$=INKEY$	X$="":IF PEEK(–16384)>127 THEN GET X$
RND(1)	Pick a random number bet ween Ø and Ø.9999.	RND(1)	RND	RND(Ø)	RND(1)
COLOR N	Set text colour for PRINT statements.	COLOUR N	INK N	REM (Computer cannot print text in colour. See footnote 1.)	REM (Computer cannot print text in colour in this resolution. See footnote 1.)

NOTE:

1. Computer ignores the word REM and carries on to the next line. It is safest to keep the line in the program using REM rather than leave it out altogether, in case there is a GOTO elsewhere in the program telling the computer to go to that line.

2. Remember to change round the numbers represented here by X and Y so that the number for Y comes first.

DOLL PROGRAM

This is the doll program that was mentioned in the second issue of the Sega User's Club Magazine. The program uses lines, circles, painting and sprites to create a rather "pretty" picture.

```
10 SCREEN2,1:CLS:COLOR 15,15,(0,0)-(255,191),15
20 SCREEN 2,2:CLS:GOSUB50:FORP=1TO1000:NEXTP:GOTO 20
30 CLS:NEXTP:GOTO20
40 REM << GIRL >>
50 RESTORE 470
60 FORL=0TO29:READ A,B,C,D:LINE(A,B)-(C,D),2:NEXTL
70 FORL=0TO48:READ A,B,C,D:LINE(A,B)-(C,D),6:NEXTL
80 PSET(114,153),1:PSET(117,153),1:PSET(116,155),1
90 LINE(87,176)-(100,176),7
100 LINE(179,168)-(198,168),7
110 LINE(225,60)-(228,77),6:LINE-(224,94),6:PAINT(227,80),6:LINE(244,112)-(244,1
25),6
120 CIRCLE(243,174),46,6,1,0.55,0.6
130 CIRCLE(231,168),32,6,1,0.6,0.72
140 CIRCLE(119,132),42,6,1,0.22,0.3
150 CIRCLE(110,138),37,6,1,0.26,0.35
160 CIRCLE(103,115),15,6,1,0.09,0.19
170 CIRCLE(99,160),16,6,1,0.35,0.52
180 CIRCLE(98,151),25,6,1,0.45,0.6
190 CIRCLE(59,110),30,6,1,0.15,0.2
200 CIRCLE(50,121),30,6,1,0,0.08
210 CIRCLE(50,121),30,6,1,0.92,1
220 CIRCLE(102,77),40,6,1,0.43,0.6
230 CIRCLE(171,108),115,6,1,0.57,0.6
240 CIRCLE(151,87),86,6,1,0.59,0.84
250 CIRCLE(125,91),105,6,1,0.865,0.948
260 CIRCLE(220,85),19,6,1,0.10,0.14
270 CIRCLE(225,119),20,6,1,0.81,0.93
280 CIRCLE(201,141),38,6,1,0.89,0.98
290 CIRCLE(205,132),30,6,1,0.91,1
300 CIRCLE(205,132),30,6,1,0,0.04
310 CIRCLE(157,138),8,6,1,0.28,0.49
320 CIRCLE(156,128),13,6,1,0.2,0.35
330 CIRCLE(180,60),49,6,1,0.30,0.45
340 CIRCLE(155,78),22,6,1,0.38,0.52
350 CIRCLE(141,68),26,6,1,0.26,0.55
360 CIRCLE(145,72),30,6,1,0.38,0.5
370 CIRCLE(130,65),29,6,1,0.28,0.45
380 CIRCLE(129,76),26,6,1,0.39,0.51
390 CIRCLE(118,72),22,6,1,0.35,0.60
400 CIRCLE(121,75),26,6,1,0.39,0.55
410 CIRCLE(100,79),13,6,1,0.25,0.47
420 GOSUB640
430 PAINT(150,50),6:PAINT(89,180),2:PAINT(94,180),2:PAINT(99,180),2
440 PAINT(171,180),2:PAINT(175,180),2:PAINT(190,180),2
450 RETURN
460 REM << GIRL LINE DATA >>
470 DATA 125,167,150,157,124,191,124,179,124,179,127,172,127,172,129,179,129,179
,130,185
480 DATA 146,191,162,171,162,171,166,163,157,191,175,173,175,173,180,166,164,191
,185,166
490 DATA 169,191,205,149,196,191,199,163,225,137,232,138,232,138,235,139,235,139
,240,143
500 DATA 240,143,245,150,245,150,250,160,250,160,255,177,250,160,254,191,99,174,
104,181,104,181,111,191
510 DATA 93,171,100,185,100,185,105,191,87,169,96,191,87,169,85,191,74,148,70,16
0,70,160,68,191
520 DATA 109,128,112,133,112,133,113,136,115,154,118,154
530 DATA 92,88,89,92,89,92,85,94,85,94,81,95,81,95,86,122,86,122,87,142,87,142,8
9,145,89,145,110,169
540 DATA 110,169,114,171,114,171,125,167,125,167,127,172,108,172,110,175,90,168,
91,173
550 DATA 91,173,87,169,84,158,80,164,80,164,76,160,80,136,75,138,75,138,69,138,7
8,109,66,92
```

```
560 DATA 225,60,224,94,224,94,227,96,227,96,231,96,231,96,237,89,237,89,236,94
570 DATA 243,112,243,126,243,126,238,135,190,166,185,166,185,166,182,165,182,165
,180,166
580 DATA 180,166,173,168,173,168,167,167,167,167,172,162,172,162,172,159,172,159
,166,163
590 DATA 166,163,161,163,161,163,157,161,157,161,161,159,161,159,158,159,158,159
,155,157
600 DATA 155,157,153,155,153,155,154,153,154,153,150,150,150,150,154,147,154,147
,154,145
610 DATA 158,141,164,136,164,136,169,129,169,129,171,124,171,124,171,115,171,115
,169,110
620 DATA 169,110,166,106
630 REM << GIRL PATTERN DATA >>
640 PATTERNS#0,"0F1F3F7FFF7FFF7E"
650 PATTERNS#1,"742C2C2C2C2E2F2F"
660 PATTERNS#2,"00F8FCFEFFFF3F1F"
670 PATTERNS#3,"0F0F0F0F0F1F3FFF"
680 PATTERNS#4,"2F2F2F2F1717130B"
690 PATTERNS#5,"0904020100000000"
700 PATTERNS#6,"F9F9FFFFFFE3E3FF"
710 PATTERNS#7,"FEFC0103FE000000"
720 PATTERNS#8,"00F8FCFEFFFF3F1F"
730 PATTERNS#9,"0F0F0F0F0F1F3FFF"
740 PATTERNS#10,"0000000000008080"
750 PATTERNS#11,"C0E0E0E06020A0A0"
760 PATTERNS#12,"F9F9FFFFFFE3E3FF"
770 PATTERNS#13,"FEFC0103FE000000"
780 PATTERNS#14,"A0A0A0A0A0A04040"
790 PATTERNS#15,"4080800000000000"
800 PATTERNS#16,"003F7FFF7FFF7FFB"
810 PATTERNS#17,"77372F2F2F2F2F2F"
820 PATTERNS#18,"7078FEFFFFFFFF9F"
830 PATTERNS#19,"0F0707030303303"
840 PATTERNS#20,"2F2F2F2727272723"
850 PATTERNS#21,"131118040300000"
860 PATTERNS#22,"87CFFFFFFFFCF8F9"
870 PATTERNS#23,"FFFFFF3F00FF0000"
880 PATTERNS#24,"000000000C0E0F8"
890 PATTERNS#25,"FCFEFEFFFFEFE7F7"
900 PATTERNS#26,"0000000000000000"
910 PATTERNS#27,"0000000000008080"
920 PATTERNS#28,"F33331F9F9F1F2E2"
930 PATTERNS#29,"E4C4881060800000"
940 PATTERNS#30,"8080808000000000"
950 PATTERNS#31,"0000000000000000"
960 PATTERNS#32,"0000000000000000"
970 PATTERNS#33,"0103070F1FFF3FFF"
980 PATTERNS#34,"0102020C1C3878F8"
990 PATTERNS#35,"F8F8F8F8F8F8F0F0"
1000 PATTERNS#36,"70FFFF0000000000"
1010 PATTERNS#40,"080C46737BFFFFFF"
1020 PATTERNS#42,"000080E0FCFFFFFF"
1030 PATTERNS#128,"003827201B0F0000"
1040 PATTERNS#129,"0000000000000000"
1050 PATTERNS#130,"0000C0380CF00000"
1060 PATTERNS#131,"0000000000000000"
1070 MAG1
1080 SPRITE0,(88,91),0,1
1090 SPRITE1,(88,107),4,1
1100 SPRITE2,(96,91),8,1
1110 SPRITE3,(96,107),12,1
1120 SPRITE4,(132,93),16,1
1130 SPRITE5,(132,109),20,1
1140 SPRITE6,(148,93),24,1
1150 SPRITE7,(148,109),28,1
1160 SPRITE20,(110,140),128,9
1170 SPRITE9,(187,153),32,2
1180 SPRITE10,(180,165),36,2
1190 SPRITE11,(86,168),40,2
1200 RETURN
```

SEGA SOUND

The last issue touched on the surface of programming sound on the Sega. The S.C. 3ØØØ has three voices. What this means is that you can play three notes all at once with different volumes. The Sega also has a noise generator that will produce white or periodic noise. One noise can be played with the music notes or chords.

To produce sound in basic we can use either the sound or beep statements.

BEEP

(See page 137 of the Level III Sega Manual).

This is a very easy way to get a simple tone. The Beep statement may have the following format:

	FUNCTION
Beep	Produces a single beep
Beep Ø	Turns sound off
Beep 1	Produces a continuous tone
Beep 2	Produces the error message tones

```
10 FOR I=1 TO 10
20 BEEP
30 NEXT I
```

Now try following changes.

(i) 2Ø Beep 1
(ii) 2Ø Beep 2
(iii) 2Ø Beep 1:Beep Ø

Another simple way to get a tone is to use the bell. This is control character 7 (please see page 18 of Level III Sega Manual). The bell can be heard by holding down the CRTL key and the [G] key.

To use the bell in a program you must print control character 7.

e.g.

```
10 FOR I=1 TO 10
20 PRINT CHR$(7)
30 NEXT I
```

SOUND

To create music or sound effects in Basic we use the sound statement with the following format:

Sound, Channel, Frequency, Volume

Channel : The channel parameter must be a number between Ø and 5.

Channel Ø: This turns all the sound off, so if you get a tone or noise playing and you cannot stop it, type

Sound Ø [CR]

Channel 1-3: These three channels allow you to create musical tones. Each channel is controlled independantly, this is channel 1 can be used for the melody while channels 2 and 3 can be used for the bass and harmony.

Channel 4: This sets the noise. Produces "white" noise.

Channel 5: This channel is periodic noise.

Frequency: This is the tone or pitch at which a note or noise is produced. For channels 1-3 the frequency can be between 11Ø-111860 hertz. If the frequency is above 111860 hertz, it is set automatically to 11Ø hertz. The human ear is unable to hear above about 2ØØØØ hertz. Certain frequencies produce notes, for example middle C is 262 hertz.

FREQUENCY TABLE						
SCALES	**f1**	**f2**	**f3**	**f4**	**f5**	**f6**
C		**131**	**262**	**523**	**1Ø47**	**2Ø94**
C#,D		**139**	**277**	**554**	**11Ø9**	**2218**
D		**147**	**294**	**587**	**1175**	**235Ø**
D#, E		**156**	**311**	**622**	**1245**	**249Ø**
E		**165**	**33Ø**	**659**	**1319**	**2638**
F		**175**	**349**	**698**	**1397**	**2794**
F#, G		**185**	**37Ø**	**74Ø**	**148Ø**	**296Ø**
G		**196**	**392**	**784**	**1568**	**3136**
G#, A		**2Ø8**	**415**	**831**	**1661**	**3322**
A	**11Ø**	**22Ø**	**44Ø**	**88Ø**	**176Ø**	**352Ø**
A#, B	**117**	**233**	**466**	**932**	**1864**	
B	**123**	**247**	**494**	**988**	**1976**	

Frequency unit Hertz.

With channels 4 and 5, the frequency is either Ø,1,2, or 3. A frequency of Ø-2 with channels 4 and 5 produces a noise at a preset frequency. A frequency of 3 produces white or periodic noise at a frequency set in channel 3.

Volume: This must be a number between Ø and 15, for all channels.

Volume Ø = Off
Volume 1 = Softest
Volume 15 = Loudest

Now that we have covered the theory we will explain what this gibberish means with the help of lots of programmes.

```
10 FOR F=110 TO 200
20 SOUND 1,F,15
30 NEXT F
40 SOUND 0
```

This program increases the frequency. This causes the pitch of the tone begin play to increase.

The frequency F is stepped from 11Ø hertz (the lowest tone the computer can make) to 2ØØ hertz in steps of 1 hertz.

Line 2Ø active sound channel 1 to play a tone with the frequency of F. F is increased by 1 in line 3Ø. When F is 2ØØ the computer goes to line 9Ø where all the sound is turned off.

Try removing line 4Ø by typing:

4Ø (CR)

Now alter line 1Ø and retype line 4Ø the program should now read:

```
10 FOR F=110 TO 400 STEP 2
20 SOUND 1,F,15
30 NEXT F
40 SOUND 0
```

Try alter the program to read

```
10 FOR F=110 TO 200 STEP 2
20 SOUND 1,F,15
30 NEXT F
40 GOTO 10
```

Channels 4 and 5 produce noises. The frequency must be a number between 1 and 3.

```
10 FOR F=0 TO 3
20 SOUND 4,F,15
30 FOR DE=1 TO 100:NEXT DE
40 NEXT F
50 SOUND 0
```

Now try channel 5 by changing line 2Ø.

```
20 SOUND 5,F,15
```

These noises still are not that much good to us as we cannot change what pitch they are played. To do this we require two sound channels. One channel is the noise generator (either channel 4 or 5) and the other is channel 3. This sets what pitch the noise is to be played at.

```
10 SOUND 4,3,15
20 FOR F=1000 TO 1500 STEP 2
30 SOUND 3,F,0
40 NEXT F
50 SOUND 0
```

What we have done is that we have turned on the sound generator (Channel 4) in line 1Ø and in line 3Ø we have altered the pitch of the piece of noise being played by changing the frequency of channel 3. The volume of channel 3 is set to Ø, so that we can hear the noise more easily. Channel 3 volume maybe between Ø and 15.

Now change the program to work with Channel 5. Make line 1Ø read

```
10 SOUND 5,3,15
```

If we use a frequency from the frequency table listed above with channels 1-3 we can get the computer to play almost any piece of music.

```
10 SOUND 1,587,15
20 FOR DE=1 TO 100:NEXT DE
30 SOUND 1,659,15
40 FOR DE=1 TO 100:NEXT DE
50 SOUND 1,523,15
60 FOR DE=1 TO 100:NEXT DE
70 SOUND 1,262,15
80 FOR DE=1 TO 100:NEXT DE
90 SOUND 1,392,15
100 FOR DE=1 TO 100:NEXT DE
110 SOUND 0
```

What we have told the computer to do in line 1Ø is to play to D. The computer then counts from 1 to 1ØØ on line 2Ø. Try removing lines 2Ø, 4Ø, 6Ø, 8Ø and 1ØØ and listening to the piece played. It is so fast that it is impossible to hear the tune. The delay loops are necessary to get a piece of music to sound as it should.

Up until now we have been playing single notes. It is possible to play up to 3 notes at once.

```
10 SOUND 1,262,10
20 FOR DE=1 TO 200:NEXT DE
30 SOUND 2,330,12
40 FOR DE=1 TO 200:NEXT DE
50 SOUND 3,392,15
60 FOR DE=1 TO 500:NEXT DE
70 SOUND 0
```

What is happening is that all the channels are playing together giving a chord.

Now let's try altering volume. The volume must be a number between Ø and 15.

```
10 FOR V=0 TO 15
20 SOUND 1,262,V
30 NEXT V
40 SOUND 0


10 FOR V=15 TO 1 STEP -1
20 SOUND 1,262,V
30 NEXT V
40 SOUND 0
```

Alter line 4Ø to read

4Ø GOTO 1Ø

In both of the above programs.

We can also alter the volume of the noise generator as well.

```
10 FOR V=0 TO 15
20 SOUND 4,2,V
30 NEXT V
40 GOTO 10
```

```
10 FOR V=15 TO 0 STEP -1
20 SOUND 4,2,V
30 NEXT V
40 GOTO 10
```

By changing both the volume and the frequency we can get some quite startling sound effects. Try the following four programs which show what can be achieved.

```
10 FOR I=0 TO 15
20 SOUND 3,5000-I*100,0
30 SOUND 5,3,I
40 NEXT
50 GOTO 10
```

```
10 FOR I=0 TO 15
20 SOUND 3,1000+I*100,0
30 SOUND 4,3,I
40 NEXT
50 GOTO 10
```

Sound Demonstrations

These three programs demonstrate the Sega audio capabilities. The first demo produces a frog jump sound effect, the second a scale and the third a piece of music using read and data statements to store the information for the piece.

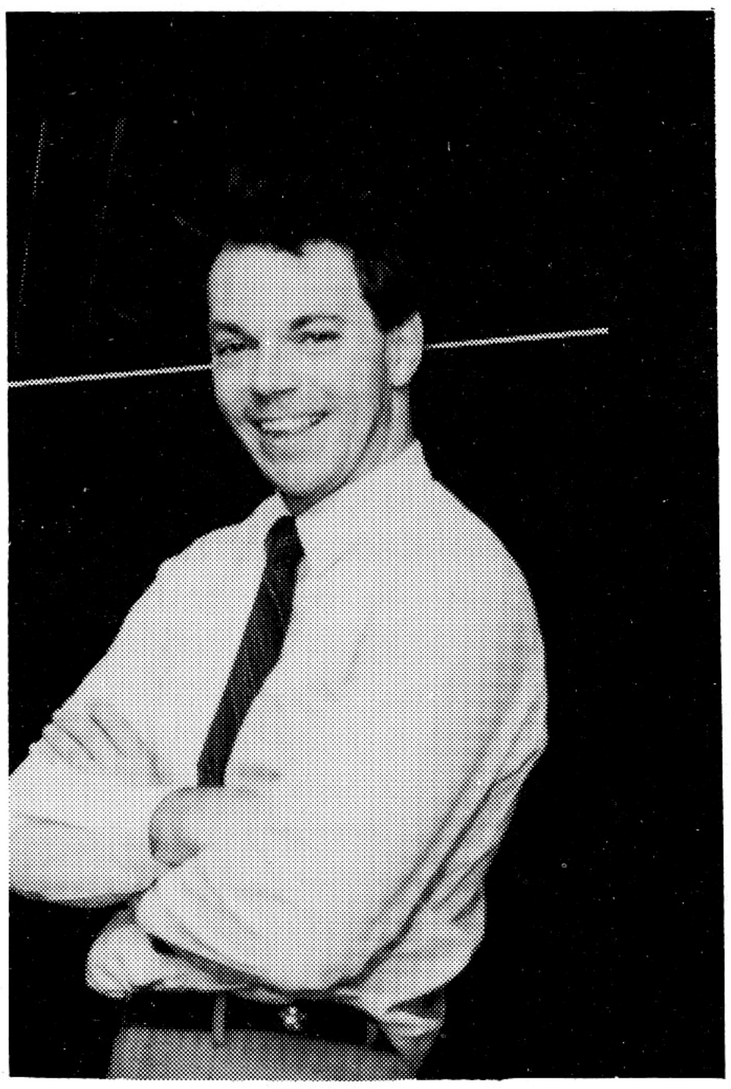
```
10 FOR I=1 TO 2
20 FOR F=500 TO 1000 STEP 100
30 SOUND 1,F,15
40 NEXT F
50 NEXT I
60 SOUND 0
```

```
20 FOR F=500 TO 750 STEP 25
30 SOUND 1,F,15
40 SOUND 2,100+F,13
50 SOUND 3,F/2,11
60 FOR T=1 TO 20:NEXT T
70 NEXT F
80 SOUND 0
```

```
10 VO=9
20 READ F2,F3,F1,D
30 IF F2=0 THEN SOUND0:END
40 SOUND 2,F2,VO:SOUND 3,F3,VO:SOUND 1,F1,VO:FOR L=1 TO D/5:NEXT:GOTO 20
50 DATA 932,1864,165,30,784,1568,311,15,932,1864,311,15,932,1864,330,15,784,1568,330,15,932,1864,330,15,784,1568,330,15
60 DATA 698,1397,349,15,932,1864,349,15,1175,2350,349,15,1397,2794,349,15,1397,2794,175,15,1175,2350,175,15
61 DATA 932,1864,175,15,698,1397,175,15,784,1568,262,240,932,1864,262,240,1175,2350,175,120,1047,2094,175
70 DATA 240,932,1864,175,120,932,1864,233,320,932,1864,175,320,932,1864,117,720,0,0,0,0
```

WHY GRANDSTAND GET RESULTS FOR SEGA

BY PHIL KENYON



Steve Pearson MD of Results Direct Marketing

Having the best computer in the country at the best price does not always mean you are going to get the most sales. Computers are not an easy product to sell. The majority of the people who buy them, know very little about them, and can easily make the wrong decision for the wrong reasons and wind up buying an obsolete or inadequate machine, purely because they recognise the brand name.

So when we at Grandstand launched a bright new star on the market, we had to make some very big decisions about spending an awful lot of time and money, in the best possible way.

The popularity of Sega, and its hugely successful launch was no accident. It is the coming together of many things:

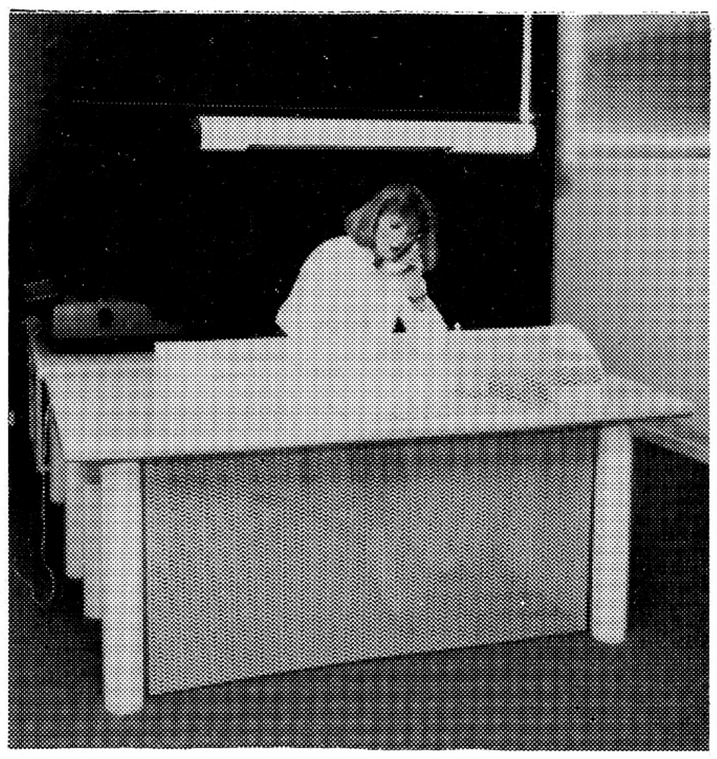
1 Years of experience in very competitive international markets to perfect marketing skills.

2 Choosing the right product mix and ensuring the product is competitive and value for money.

3 Creating consumer awareness to the right people at the right time, in the best way.

Whenever we decide on a campaign we need to move quickly to keep pace with the trends and to get results.

Our first phone call is to Results Direct Marketing in Parnell, to discuss our idea. The conversation usually goes something like:



Kerri Citanovich, always a cheerful welcome

Good Morning Results . . . . . Oh hi Steve . . . . . Sorry Phil (you poms all sound the same). I will put you through to Steve Pearson.

Steve Pearson speaking . . . . . Hi Steve how are you . . . . . , Sorry Phil!. What can I do for you . . . You want to do some public seminars. Ok, let's get together tomorrow and discuss the format, when are you planning to start them . . . . . You are kidding! . . . . . Roger and I will be around in an hour. Bye!

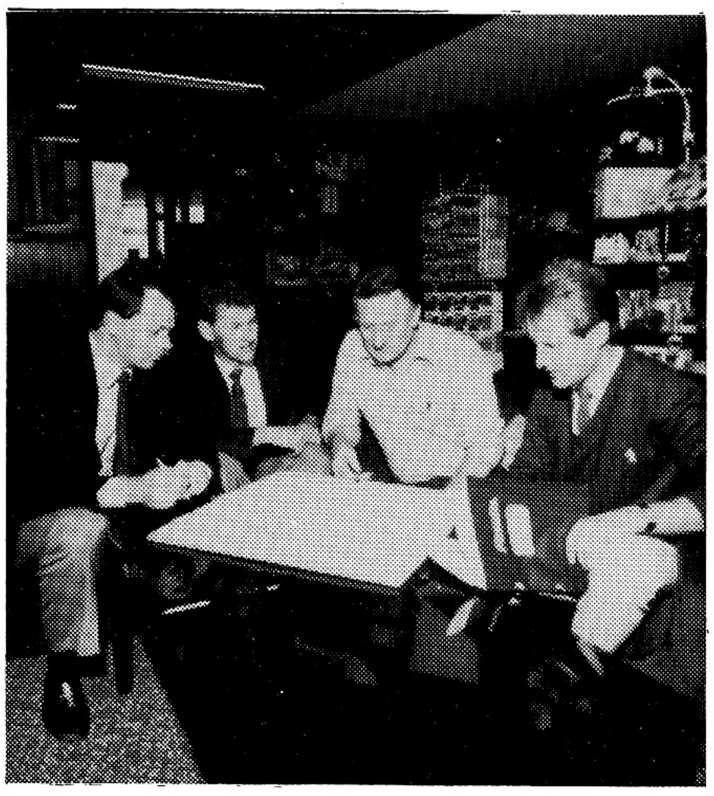
Results understand that when you are dealing with electronics products, you have got to keep pace with what the public is looking for and with computers it's knowledge. How to overcome the barrier of understanding something completely new, without spending so much money on advertising it that the consumer cannot afford to buy it, is a very important area, requiring specialised skills and knowledge.

We have documented a typical campaign to give you an insight on the fascinating behind the scenes preparations.

The first meeting usually involves Les Kenyon, our Sales Director, Steve Pearson . . . . . Roger McMillian and myself. Here we discuss the type of interest we want to generate, the features we feel should be stressed most, bearing in mind current public attitudes, any potential drawbacks which we must overcome, and the areas which we need to concentrate our activities.

Results then go to the drawing board to produce a storyboard type presentation. Armed with information which to the normal mortal would conjure up images of dismembered bodies avidly watching T.V. (only .6 of a person in each house watches underwater Hockey each week), they can quite accurately assess an advertising campaign which will reach the areas of the public who are most likely to be interested in the information we have to give.

A Breakdown of the costs of production for the campaign and the advertising which would be required in order for it to be successful and the expected response to that advertising is then laid out, and presented to the Managing Director for further discussion. As Managing Director, Bill Fentons' primary task is to ensure all of the companies interests are being best served by each campaign, whereas, specialising in one area can result in "tunnel vision" many other aspects of the company can also be represented of the company by subtle inclusions and overall foresight at this time.



Left to right: Steve, Roger, Les, Phil, talking tactics

The timing of any campaign which is likely to increase demand is crucial. It must tie in with many areas. Schools and public holidays must be considered.

Reasonable warning must be given to dealers in order that they have sufficient

Bill Fenton, crosses some T's with Steve and Roger

stock to supply demand, our own stocks must be sufficient with ongoing supplies catered for, also the financing for all of this planned well in advance.

Once a campaign is scheduled and approved, then everyone must work quickly. Results "Art Director" has now been called in to produce layouts and copywriters set out to make sense of difficult terminology and to give what could otherwise be uninteresting matter, a readable and compelling flavour. The campaign must not only create awareness but also show the useful relevant features of the product, and produce a certain amount of urgency to learn more.

With Sega, it was decided that the most effective way of promoting the computer was to get face to face with as many people as possible and to tell a story.

The story had to be factual, and informative, without being confusing and also light hearted enough to hold peoples attention and be amusing.

Eventually a complete story is outlined and agreed upon in comic strip form, this is then taken by Results to be made up into a series of slides, to be used at public seminars, which will be arranged all around the country in the hope of showing people who may otherwise be too confused to understand how simple and useful the Sega is to operate, and the many related benefits.

Behind the scenes there is still a great deal of work to be done coordinating a tour of seminars requires a strict schedule of dates, coupled with an effective journey plan, which will involve all of the Grandstand sales force, in each area. Suitable venues must be arranged and the concept must be sold to each local dealer, whose assistance will be required.

Karen Meldrum and Paul piece it all together

To do this a package must be produced to promote the seminar in each area, to ensure a worthwhile attendance.

Karen who is Studio Director with results now sets to work with her scalpel, to engineer attractive artwork, which can be used to advertise each individual evening, once again the wording for each ad is carefully chosen to create interest and catch the eye of the people we are wanting to attend.

Publications are selected in each area, in which the adverts will be placed.

The copy for ads is produced as a form of photograph, along with Sega logos, and it is this which Karen cuts out and arranges around any graphic designs or

Paul Hooker (Art Director) joins the team to add his thoughts and direction

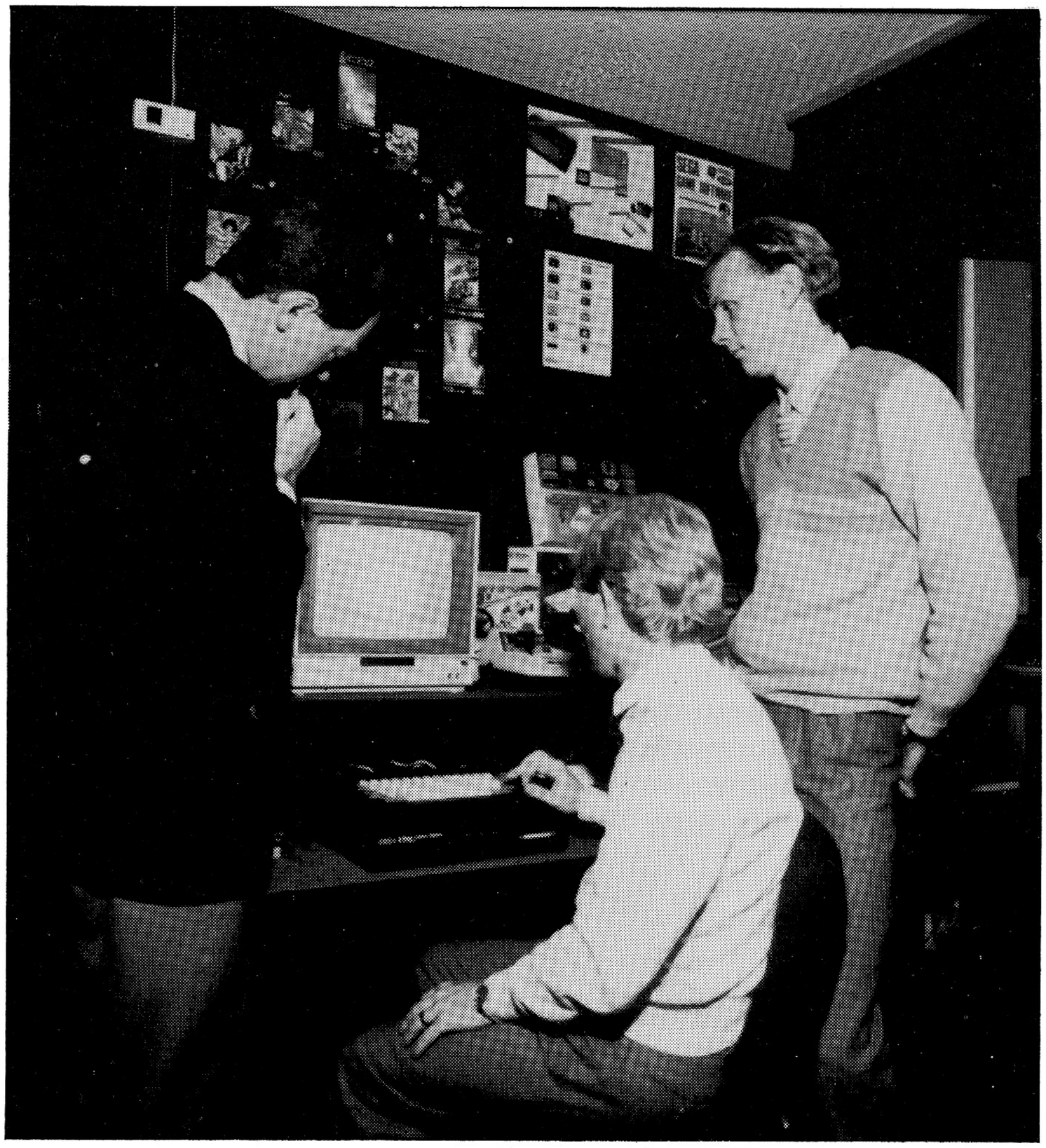
photographs which will accompany the ad.

This is then despatched to each chosen paper, and space is booked prior to each proposed seminar, with the "Where and When" suitably included. Tickets and Window posters complete the package once again, all suitably inscribed to attract interested parties.

The electronics industry changes pace and direction so quickly it is vital that a market strategy can be arrived at quickly and acted on instantly. New ideas must constantly be introduced and if possible boast the awareness of other products in the range such as the Grandstand Electronics.

The three areas of "Marketing Sales & Product Knowledge" are very closely linked, therefore, Results have found that they have had to become familiar with our Sales Techniques, in all areas of the range and also they have had many meetings with our technicians for computer training, conversely it is not at all unusual for ourselves to sit and write copy for our ads. The ability to work as one team is one of the factors of the current success.

As Grandstand continues to increase its market share in al. areas of home electronics and continues to introduce new and innovative products, we will also continue to be as innovative in the ways in which we have our products marketed, in order to continue to be successful.



Mark Vercoe the Sales Technician (right) and Steve Kenyan electronic games Sales Manager (seated), keep Steve Pearson up to date on the product range.

SEGA SUPERSTAR

HIGH SCORE TABLE

A new regular feature beginning in your next issue of the Sega User Magazine, will be a top 5 score table for all Sega game cartridges.

The player recording the highest points tally each month will receive a free cassette program, and have their name published in the Magazine, along with their score and the scores of the four runners up.

To have your score entered, we must have some form of verification as to its validity, ideally a photograph of the screen, showing the recorded high score. In games where the high score may be flashing, the screen action can be frozen, by depressing the Reset key.

HIGH SCORES

Congratulations to 13 year old Stephen Winter of Christchurch in a feat probably only rivalled by Peter Brock in James Hardy 1000.

Stephen has chalked up an incredible 999,999 in Monaco GP which does not leave an awful lot of room for improvement (unless we introduce a new blindfold category).

Well done Stephen and I hope the blisters on your fingers from all that driving are soon better.

Philip Kenyon

JINGLE BELLS

With the festive season upon us once again here is a small tune to get you into the Christmas mood.

Notice line 6Ø: Here information is read from the data listed in lines 1ØØ inwards. It will not be necessary to retype all the lines as some to them are the same. Just change the line number.

```
 10 REM ****************
 20 REM * JINGLE BELLS *
 30 REM ****************
 33 COLOR 15,4
 35 CLS:CURSOR 13,23:PRINT "JINGLE BELLS"
 37 FOR I=1 TO 12:PRINT:NEXT I
 40 RESTORE:V=12
 50 FOR X=1 TO 223
 60 READ F1,F2,F3,DE
 70 IF F1=0 THEN SOUND0:NEXT X
 80 SOUND 1,F1,V:SOUND 2,F2,V:SOUND 3,F3,V:FORD=1TODE:NEXTD,X
 85 SOUND 0
 90 FOR D=1 TO 250:NEXT D:GOTO 40
100 DATA 880,698,262,20,0,0,0,0,880,698,440,20,0,0,0,0,880,698,392,20,0,0,0,0,880,698,349,20,0,0,0,0
110 DATA 880,698,262,20,0,0,0,0,880,698,262,20,0,0,0,0,880,698,262,20,0,0,0,0,880,698,262,20,0,0,0,0
120 DATA 880,698,262,20,0,0,0,0,880,698,440,20,0,0,0,0,880,698,392,20,0,0,0,0,880,698,349,20,0,0,0,0
130 DATA 932,784,294,20,0,0,0,0,932,784,294,20,0,0,0,0,932,784,294,20,0,0,0,0,932,784,294,20,0,0,0,0
140 DATA 932,784,294,20,0,0,0,0,932,784,466,20,0,0,0,0,932,784,440,20,0,0,0,0,932,784,392,20,0,0,0,0
150 DATA 932,784,330,20,0,0,0,0,932,784,330,20,0,0,0,0,932,784,330,20,0,0,0,0,932,784,330,20,0,0,0,0
160 DATA 932,784,523,20,0,0,0,0,932,784,523,20,0,0,0,0,932,784,466,20,0,0,0,0,932,784,392,20,0,0,0,0
170 DATA 880,698,440,20,0,0,0,0,880,698,440,20,0,0,0,0,880,698,440,20,0,0,0,0,880,698,440,20,0,0,0,0
180 DATA 880,698,262,20,0,0,0,0,880,698,440,20,0,0,0,0,880,698,392,20,0,0,0,0,880,698,349,20,0,0,0,0
190 DATA 880,698,262,20,0,0,0,0,880,698,262,20,0,0,0,0,880,698,262,20,0,0,0,0,880,698,262,20,0,0,0,0
200 DATA 932,784,294,20,0,0,0,0,932,784,466,20,0,0,0,0,932,784,440,20,0,0,0,0,932,784,392,20,0,0,0,0
210 DATA 932,784,294,20,0,0,0,0,932,784,294,20,0,0,0,0,932,784,294,20,0,0,0,0,932,784,294,20,0,0,0,0
220 DATA 932,784,294,20,0,0,0,0,932,784,466,20,0,0,0,0,932,784,440,20,0,0,0,0,932,784,392,20,0,0,0,0
230 DATA 932,784,523,20,0,0,0,0,932,784,523,20,0,0,0,0,932,784,523,20,0,0,0,0,932,784,523,20,0,0,0,0
240 DATA 932,784,523,20,0,0,0,0,932,784,523,20,0,0,0,0,932,784,466,20,0,0,0,0,932,784,392,20,0,0,0,0
250 DATA 880,784,349,50,0,0,0,0,523,330,233,50,0,0,0,0
260 DATA 440,349,175,20,0,0,0,0,440,349,262,20,0,0,0,0,440,349,131,25,440,349,262,20,0,0,0,0
270 DATA 440,349,175,20,0,0,0,0,440,349,262,20,0,0,0,0,440,349,131,25,440,349,262,20,0,0,0,0
280 DATA 440,349,175,25,523,440,262,25,349,131,233,36,392,262,30000,12
290 DATA 440,349,175,25,440,349,262,25,440,349,131,25,440,349,262,25
300 DATA 466,294,233,20,0,0,0,0,466,294,175,20,0,0,0,0,466,294,123,20,0,0,0,0,466,294,123,20,0,0,0,0
310 DATA 466,349,175,20,0,0,0,0,440,262,131,20,0,0,0,0,440,262,175,20,0,0,0,0,440,262,175,20,0,0,0,0
320 DATA 440,247,196,20,0,0,0,0,392,247,147,20,0,0,0,0,392,247,196,20,0,0,0,0,440,247,147,20,0,0,0,0
330 DATA 392,330,131,25,392,330,147,25,523,392,165,25,523,392,131,25
340 DATA 440,349,175,20,0,0,0,0,440,349,262,20,0,0,0,0,440,349,131,25,440,349,262,20,0,0,0,0
350 DATA 440,349,175,20,0,0,0,0,440,349,262,20,0,0,0,0,440,349,131,25,440,349,262,20,0,0,0,0
360 DATA 440,349,175,25,523,440,262,25,349,131,233,36,392,262,30000,12
370 DATA 440,349,175,25,440,349,262,25,440,349,131,25,440,349,262,25
380 DATA 466,294,233,20,0,0,0,0,466,294,175,20,0,0,0,0,466,294,123,20,0,0,0,0,466,294,123,20,0,0,0,0
390 DATA 466,349,175,20,0,0,0,0,440,262,131,20,0,0,0,0,440,262,175,20,0,0,0,0,440,262,175,20,0,0,0,0
400 DATA 523,330,131,20,0,0,0,0,523,330,147,20,0,0,0,0,466,294,165,20,0,0,0,0,392,247,131,20,0,0,0,0
410 DATA 349,220,175,25,349,220,131,25,698,440,175,50
```

Champion Tennis

Serve, receive and smash. Pit your skill against the computer or a friend in a game of tennis of superb realism. Attack the rival's weak point. Control each shot as he or she plays forehand or backhand, volleys, lobs, drop shots and moves around on the 3D court.

This is a highly intellectual and competitive sports game which requires sharp nerves and concentration to get a victory over your rival.

N-sub

You are the Captain of a powerful submarine under attack from all sides. Fire a torpedo when you find the enemies' ships and destroy enemy submarines before they destroy you. Avoid the battleship and destroyers above as they attack with missiles, torpedoes and depth charges. Counter attack and sink the enemy ships. The enemy attack becomes more intensive with each new round.

Borderline

This is a game in which you must manoeuvre your jeep through enemy fields. This multi-screen action game requires you to move your jeep in all directions to fight the enemy by taking full advantage of shooting in both the left and right directions. Lead to the enemy to be lost in a maze using intelligent tactics.

After destroying all the fortresses you will eventually reach the heavily protected headquarters which you must destroy. A thrilling and exciting action game!!

Yamato

Avoid the enemy's attack by torpedoes and guided missiles. Cruise across the mighty ocean on this fearsome battleship while destroying the enemy vessels with your torpedoes and rockets. Be careful as Yamato could explode if hit by a guided missile or torpedo. Aim your sights at the enemy and fire the guns quickly to destroy the submarines, ships and planes. Can Yamato survive through this fierce battle?

SOFTWARE REVIEW

Congo Bongo

Big arcade game fun as you, the hunter, climb up the treacherous cliffs, jumping rivers and waterfalls, dodging flying coconuts and avoiding the poisonous snakes. Among the many dangers you encounter on the way are the pesky monkeys which slow you right down.

The aim of the game is to catch the gorilla to preserve your hunters' honour.

Pacar

A mischievous car runs through the 3-dimensional roads advancing and reversing at high and low speed. Try to eat up all the dots in the screen while avoiding two forms of enemy cars. When you drive over a power dot Pacar becomes invincible as it increases in power and will be free from damage when it crashes into the enemy car.

Go as fast as you can and destroy as many enemy cars as possible. However, Pacar's task is not easy as as soon as he has driven over all the dots they re-appear and there are more enemy cars to chase him.

Champion Baseball

3-D realism as you play to win against the computer. Swerve that pitch steal base, swing the bat and chase the fly balls. You can enjoy all imaginable kinds of baseball plays one after another as in the actual game. Double plays and the elusive home run are all possible on this great sport cartridge.

Pop flamer

A fast and furious multi-screen game featuring a flame throwing mouse. You attempt to break balloons and supply fuel for your flamethrower while incinerating the four forms of monster which are doing their best to kill you. However, be careful of the paralysing ray form the purple "HARADON." When you are hit by this beam you are unable to move for a short period of time until its affects wear off. When POP FLAMER takes a drink from the power well at the top and bottom of the maze he turns into "SUPER MOUSE," swallows the monsters nearby and speeds off to victory!

Champion Golf

If it's raining on Sunday . . . it's "CHAMPION GOLF" day. This game is probably the most challenging and realistic golf game ever devised for a computer. You can enjoy walking around the nine hole golf course with considerate caddie over its extensive fairways, roughs, greens, ponds and bunker areas. The changing weather conditions and the trees and enormously to the realism and play value of this marvellous game. You can even get a "Birdie," "Eagle" or an amazing "Hole-in-one." An absolute must for any sports enthusiast.

Exerion

An excellent conversion of this very popular new arcade game, which puts you in control of the FIGHTER EX which rushes in and out attacking the waves of enemy craft. There are six forms of alien craft to be destroyed. See how successful you are in battling the enemy through the six dangerous worlds. Certain to be one of the most popular games for SEGA ever.

Video Flipper

Turn your screen into a challenging pin-ball machine. A colourful exciting game involving split second timing. All the features of a real arcade pin-ball machine with catchy realistic sound effects and bonus pockets. Take good aim and shoot away on this game which is probably the most exciting pin-ball game you are ever likely to see on your own T.V. A must for every pin-ball wizard.

Star Jacker

Pilot your fleet of fighter ships through space while avoiding the enemy attackers. Make as many runs towards the enemy star base as you can weaving left, right and up and down to avoid the enemy fire. If you destroy the master alien you will warp to the mother ship to be re-fuelled and get reinforcements.

This is a true arcade action game that gets progressively harder and harder for every second you survive.

SOFTWARE REVIEW

Monaco GP

A red formula racing car travels at high speed through the streets speeding up and slowing down to left and right to avoid the cars obstructing your way. However, if the other cars come too close to your valued machine you can take off and jump right over the top of them. Be on the lookout for the ambulance passing by to the aid of less skilful drivers and keep an eye out for the sign posts!!!

As your car races through the winding circuit you must pit your wits against mind boggling obstacles, including slip zones, dangerous construction sites, sharp curves and narrow bridges which may also appear.

When in the tunnel the headlights of your car allow you to see a short distance so be careful. After improving your skill by going through level 1, the beginners' level, you can move on to the more dangerous levels, 2 and 3.

Safari Hunting

Go deep down into the jungle and hunt fierce animals. Your weapon is a tranquiliser gun which allows you to capture sleeping snakes, gorillas, lions and elephants to put in your truck. Stalk your prey, using instincts of the big game hunter, taking care to aim well at your prize. Beware the jungle is full of dangerous surprises.

Sindbad Mystery

Sindbad is an adventurous boy who lands on Treasure Island and is searching to find the artefacts hidden there. The exact location of the treasure becomes clearer and clearer. As Sindbad searches through the 3D maze his only friend is his trusted rock which he can kick and roll to destroy the monsters.

Can Sindbad successfully find the buried treasure?

INDY GP

BY ANDREW FLAXMAN

This is a very simple version of your favourite game Monaco GP. The program itself is of basic design so you can add and improve as you progress.

The game involves you as a Grand Prix car driver who hopefully dodges the onslaught of cars coming your way. By dissecting this game you will learn how to use the infamous 'sprite' graphics of the Sega and also many other commands that other computer owners can only dream of.

Line 1Ø:-

As mentioned in the last issue the REM statement is used as a reminder when listing your program of what's where. It has no effect on the outcome of your program.

Line 2Ø:-

This simply changes the shape of character number 44 (which happens to be a 'comma') into a racing car for showing how many cars we have left.

Line 3Ø:-

SCREEN 2,2 puts your Sega into graphics mode and CLS simply clears the screen. The following command puts sprite number Ø at 127 pixels or dots across and 175 pixels down. It then assigns pattern Ø and colour 1Ø to the sprite. Mag 2 assigns on 8 by 8 grid sprite which is magnified to be a 16 × 16 sprite We end up with a "blocky" 16 by 16 car.

Line 4Ø:-

This sets the variables CA to Ø. CA is how many cars we have past.. A=RY is the position of your car. Me=3, which is how many men or cars you start with, and J=12 which is the speed of the oncoming cars.

Line 5Ø:-

Positions the three cars at top or right of screen.

Line 6Ø:-

Prints 'cars past' at left of screen.

Line 7Ø-8Ø:-

Assigns patterns or shapes for cars and crash explosion.

Line 9Ø:-

Sets the variable I as being from Ø to 255 in steps of 16. C is a random number that changes colours in the colour statement. The colour statement draws multicoloured lines as your background. Next I tell the computer to repeat the task.

Line 1ØØ-11Ø:-

Puts a white background behind screen writing to it is easily read.

Line 12Ø:-

Uses COLOR statement to draw the actual road.

Line 13Ø:-

Another REM which let's us know we are now into the main loop.

Line 14Ø-15Ø:-

SP which is the speed of the other cars. V and DV determine what angle the opposing cars come down on. X is a random placing of the opposing car at the top.

Line 16Ø-17Ø:-

Increases the speed by increments of .4 pixels everytime a new car arrives. The variables give and angle of decent. Line 17Ø is a RND number between Ø-14 with 2 added. This randomly puts a colour to the baddies car and border colour.

Line 18Ø:-

Assigns the colour and patterns to sprite Ø and 1. The second statement uses a noise channel 5, frequency 2, volume 15, which sounds rather like a sick engine.

Line 19Ø:-

We are now right in the thick of it. This is what is commonly known as the joystick loop.

Checks if the stick (1) is equal to 3. That is if you are pushing your joystick to the right. As the manual explains the joystick has eight directions.

1=up, 2=up-right, 3=right, 4=down-right, 5=down, 6=down-left, 7=left and 8=up-left.

Therefore, if we are pushing the joystick to the right it will equal 3 and add 5 on to A which will move A (X axis of Sprite Ø 'you') to the right.

Line 2ØØ:-

Exactly the same as line 19Ø but moves Sprite Ø 5 spaces to left.

Line 21Ø:-

Checks to see you go off either side of road.

Line 23Ø:-

Tells the computer to go back to 19Ø. 19Ø is the beginning of our joystick loop.

Line 24Ø:-

Remember from line 21Ø, it said to go to line 24Ø. If sprite 1 (oncoming car) is past bottom of screen. It then checks coordinates of you, and opposing car. i.e. X+16 is bigger than A and X is smaller than A+16, then crash which is on line 29Ø. If the coordinate does not match up then it will bypass and carry onto the next line number 25Ø.

Line 25Ø:-

Another REM telling us this is the beginning of printing how many cars we have past.

Line 26Ø:-

SPRITE 1, (2ØØ,2ØØ) simply puts sprite 1 at locations 2ØØ across and 2ØØ down is below the border. The BLINE statement clears away the last number of cars past number and makes way for a new number.

CA=CA+1 adds another car onto our total of cars passed.

Line 27Ø:-

By printing CHR$(17) doubles the size of characters on.SCREEN 2,2. COLOR 5 changes the writing colour to light blue. It will now print at location 2Ø by 7Ø down in a blue colour and double size how many cars have been passed.

Line 28Ø:-

Tells the computer to go to 14Ø. This will again randomly speed and coordinates for oncoming cars etc. In effect it is starting again.

Line 29Ø:-

By printing CHR$(16) it returns character size to normal. Now it makes I equal from 15 to Ø in a step of-:5 This routine sets the sound channel I and noise channel 5 on and uses I as the decreasing volume to sound like a crash or explosion. It also changes the pattern and shape of Sprite Ø.

Line 3ØØ:-

Now we have crashed we must subtract 1 from our cars remaining if any. We

now must clear the cars and print up how many we have left.

Line 31Ø:-

Check if we have less than 1 car left and if so go to line 38Ø which is our game over routine.

Line 32Ø:-

Checks if ME (MEN) is two then print two cars at location 2Ø8 across by 4Ø down. Remember the commas were redefined as cars.

Line 33Ø:-

Checks if 1 car is left and if it is it prints one car.

Line 34Ø:-

Another GOTO 14Ø.

Line 35Ø:-

Remember line 22Ø where it checked to see if your car was off the side of the road. If your car was then it said to goto 35Ø.

It now checks if A is greater than 85 then A will equal 85 so your car goes no further off the left hand side of the road. Goto 19Ø resumes play.

Line 36Ø:-

Checks to see we are not leaving the right hand side of the road.

Line 38Ø:-

This is the game over routine. We now print 'strike a key to start' at position 85, 9Ø with a colour of purple.

Line 39Ø:-

Prints "The End" at position 195,4Ø.

Line 4ØØ:-

Finds a random number for colour changing. It then changes the colour adding 1 as not to have a transparent colour and then changes the border colour.

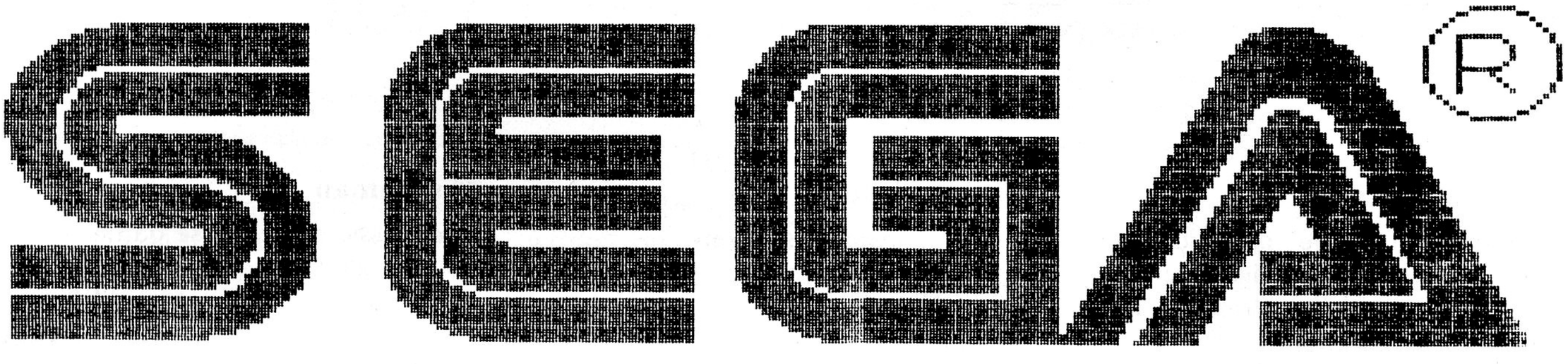
Line 41Ø:-

GOTO 39Ø. Simply keeps scanning the keyboard and changing the colours.

This program is very basic so you can add many more exciting and interesting variations into the game.

```
10 REM <*<*<INDY GP>*>*>
20 PATTERN C#44,"5AFF5A1824BDE799"
30 SCREEN 2,2:CLS:SPRITE0,(127,175),0,10:MAG2:PRINTCHR$(16)
40 CA=0:A=127:ME=3:J=12
50 CURSOR200,40:COLOR1:PRINT ",  ,  ,"
60 CURSOR13,50:PRINT"CARS PASSED"
70 PATTERNS#0,"5AFF5A1824BDE799"
80 PATTERNS#1,"1981821617165751"
90 FORI=0TO255STEP8:C=RND(1)*16:COLOR,C,(I,0)-(I,191),C:NEXTI
100 COLOR15,15,(10,45)-(80,80)
110 COLOR15,15,(195,35)-(235,50)
120 COLOR1,1,(80,0)-(175,191),5
130 REM <SPRITE MOVEMENT>
140 V=INT(RND(1)*3)
150 DV=INT(RND(1)*3):X=(RND(1)*95)+80
160 J=J+.4:FV=V-DV
170 C=INT(RND(1)*14)+2:COLOR,,,C
180 SP=10:SPRITE1,,0,C:SPRITE0,,0,10:SOUND5,2,15
190 IFSTICK(1)=3THENA=A+5:GOTO210
200 IFSTICK(1)=7THENA=A-5
210 X=X+FV:SP=SP+J:SPRITE0,(A,167):SPRITE1,(X,SP):IFSP+19>180THEN240
220 IFA<85ORA>155THEN350
230 GOTO190
240 IFX+16>AANDX<A+16THEN290
250 REM PRINT CARS PASSED
260 SPRITE1,(200,200):BLINE(10,70)-(55,79),,BF:CA=CA+1
270 PRINTCHR$(17):COLOR5:CURSOR20,70:PRINTCA:GOTO140
280 GOTO140
290 PRINTCHR$(16):FORI=15TO0STEP-.5:SPRITE0,,0,6:SOUND1,110,I:SPRITE0,,1,
D4,0,I:NEXTI
300 ME=ME-1:BLINE(200,40)-(235,50),1,BF
310 IFME<1THEN380
320 IFME=2THENCURSOR206,40:PRINT",  ,"
330 IFME=1THENCURSOR206,40:PRINT","
340 GOTO140
350 IFA<85THENA=85:GOTO190
360 IFA>155THENA=155:GOTO190
370 GOTO140
380 COLOR13:CURSOR85,90:PRINT"STRIKE A KEY TO
"
390 CURSOR195,40:PRINT"THE END"
400 C=RND(1)*14:COLORC+2:COLOR,,,C
410 IF INKEY$<>""THEN10
420 GOTO390
```

GRAPHICS DEMONSTRATION




```
10 SCREEN 2,2:COLOR 1,15,(0,0)-(9,9),15:CLS:POSITION(13,0),0,0
20 LINE(30,70)-(3,70),4
30 LINE-(3,63)
40 LINE-(30,63)
50 LINE(30,61)-(3,61),4
60 LINE-(3,54),4
70 LINE-(30,54),4
80 LINE(20,50)-(30,50),4
90 LINE (20,43)-(30,43),4
100 LINE (20,41)-(30,41),4
110 LINE (20,34)-(30,34),4
120 LINE (20,30)-(47,30),4
130 LINE -(47,23),4
140 LINE -(20,23),4
150 LINE (20,21)-(47,21),4
160 LINE -(47,14),4
170 LINE -(20,14),4
180 CIRCLE (20,32),2,4,1,.25,.75
190 CIRCLE (20,32),9,4,1,.25,.75
200 CIRCLE(20,32),11,4,1,.25,.75
210 CIRCLE(20,32),18,4,1,.25,.75
220 CIRCLE(30,52),2,4,1,.75,.25
230 CIRCLE(30,52),9,4,1,.75,.25
240 CIRCLE(30,52),11,4,1,.75,.25
250 CIRCLE(30,52),18,4,1,.75,.25
300 LINE(75,70)-(105,70),4
310 LINE-(105,63),4
320 LINE-(75,63),4
330 LINE(75,61)-(105,61),4
340 LINE-(105,54),4
350 LINE-(79,54),4
360 LINE(75,50)-(100,50),4
370 LINE-(100,43),4
380 LINE-(68,43),4
390 LINE-(68,53),4
400 LINE(68,30)-(68,41),4
410 LINE-(100,41),4
420 LINE-(100,34),4
430 LINE-(75,34),4
440 LINE(79,30)-(105,30),4
450 LINE-(105,23),4
460 LINE-(75,23),4
470 LINE(75,21)-(105,21),4
480 LINE-(105,14),4
490 LINE-(72,14),4
500 CIRCLE(75,30),7,4,1,.5,.75
510 CIRCLE(75,30),9,4,1,.5,.75
520 CIRCLE(75,30),16,4,1,.5,.75
530 CIRCLE(75,54),7,4,1,.25,.5
540 CIRCLE(75,54),9,4,1,.25,.5
550 CIRCLE(75,54),16,4,1,.25,.5
560 LINE(59,30)-(59,54),4
570 LINE(66,30)-(66,54),4
580 CIRCLE(79,34),4,4,1,.5,.75
590 CIRCLE(79,50),4,4,1,.25,.5
600 LINE(129,70)-(160,70),4
610 LINE-(160,34),4
620 LINE-(133,34),4
630 LINE-(133,41),4
640 LINE-(153,41),4
650 LINE-(153,63),4
660 LINE-(129,63),4
670 LINE(129,61)-(151,61),4
680 LINE-(151,43),4
690 LINE-(133,43),4
700 LINE-(133,50),4
710 LINE-(144,50),4
720 LINE-(144,54),4
730 LINE-(130,54),4
740 LINE(128,14)-(160,14),4
750 LINE(128,21)-(160,21),4
755 LINE-(160,14),4
760 CIRCLE(130,32),2,4,1,.5,.75
770 CIRCLE(128,30),7,4,1,.5,.75
780 CIRCLE(128,30),9,4,1,.5,.75
790 CIRCLE(128,30),16,4,1,.5,.75
800 CIRCLE(130,52),2,4,1,.25,.5
810 CIRCLE(128,54),7,4,1,.25,.5
820 CIRCLE(128,54),9,4,1,.25,.5
830 CIRCLE(128,54),16,4,1,.25,.5
840 LINE(128,23)-(160,23),4
850 LINE-(160,30),4
860 LINE-(130,30),4
870 LINE(128,32)-(128,52),4
880 LINE(121,30)-(121,54),4
890 LINE(119,30)-(119,54),4
900 LINE(112,30)-(112,54),4
1000 LINE(185,21)-(160,70),4
1010 LINE-(169,70),4
1020 LINE-(189,30),4
1030 LINE(192,30)-(172,70),4
1040 LINE-(181,70),4
1050 LINE-(195,42),4
1060 LINE-(201,54),4
1070 LINE-(191,54),4
1080 LINE-(191,61),4
1090 LINE-(213,61),4
1100 LINE-(198,30),4
1110 LINE(202,30)-(217,63),4
1120 LINE-(191,63),4
1130 LINE-(191,70),4
1140 LINE-(230,70),4
1150 LINE-(206,20),4
1160 CIRCLE(195,25),11,4,1,.59,.92
1170 CIRCLE(195,33),4,4,1,.6,.9
1171 CIRCLE(195,33),7,4,1,.6,.91
1180 PAINT(25,18),4:PAINT(25,65)
     ,4:PAINT(80,18),4:PAINT(80,25),4
1181 PAINT(80,45),4:PAINT(150,16)
     ,4:PAINT(150,25),4
1182 PAINT(195,20),4:PAINT(195,35),4
1190 CIRCLE(225,20),10,4,1,0,1
1200 LINE(225,15)-(220,15),4
1210 LINE-(220,25),4
1220 CIRCLE(225,18),3,4,1,.75,.25
1230 LINE(225,21)-(228,25),4
1240 LINE(220,21)-(225,21),4
1250 GOTO 1250
```

CASSETTE SOFTWARE EDUCATION

Learning Alphabet

Picture book type graphics teach children the alphabet in a fun way, designed to hold the attention longer with bright colours, movement and sound.

Learn to Count

Number recognition and elementary counting to prepare young minds for basic mathematics.

Shape & Colour Quiz

Random objects in varying colours and numbers must be recognised and counted. Ideal for shape and colour recognition and for learning to add.

Addition Subtraction Multiplication

Three separate programs with one common theme. A classroom teacher who puts forward problems as they would be presented in the classroom. The computer will give help when asked in solving the problem, and will help when incorrect answers are given. 5 difficulty levels take children from age 5 to 12 through maths which is fun to do on a computer, and is bound to assist in this subject in school.

Spelling

This program requires you to enter in a series of words which the computer will flash on the screen briefly at random, to be copied by the child learning to spell. This enables the parent or teachers to tailor the level of difficulty to suit all ages, and ability.

Rocket Maths

Select problem range, either ADDITION, SUBTRACTION, DIVISION, MULTIPLICATION. Difficulty level 1Ø to 2Ø, then solve the problems on the screen while guiding your rocket ship across the screen into one of the six ports showing a solution, only one of which is correct. It's a race against time to solve six problems as quick as possible to record your name on the top 1Ø high score board (ages 7-14).

Watch Me Draw

Control a paintbrush on the screen by using the Joystick, dip into the different colour pots to change colours. Hours of fun doodling and drawing on your T.V. screen.

Typing Tutor

See applications software.

Empire

See entertainment. Teaches decision making and Government.

Music Editor

Shows how to use sound and music commands.

Sprite Editor

Assists in creating shapes and patterns to animate your programs.

Teach Yourself Basic Games Programming

An excellent package consisting of an extensive book and cassette, which clearly demonstrates all aspects of control, sound and animation as well as the elementary mathematics required to create your own exciting programs.

Mars Mobile

Arcade type action game to manoeuvre across a planet's surface shooting down attackers from above.

Towers of Hanoi

A mind challenger used by ancient monks in Hanoi to move rings of varying size from one pole to another in a set sequence in the least number of moves. Challenge the computer to a battle of wits. With nine levels of difficulty.

Hangman

The classic hangman word challenge featuring three difficulty levels and computer graphics. Lots of fun and educational value.

City Lander

A game of skill which involves controlling a descending spacecraft through a variety of challenging paths. In order to land on the power platforms to score points, speed is essential to keep up fuel supplies.

Cube-it

A version of the popular arcade game, controlling a hopping creature whose object is to land on each block, changing its colour and avoiding capture. Seventeen screens of increasing difficulty are incorporated in this very challenging game.

Munchman

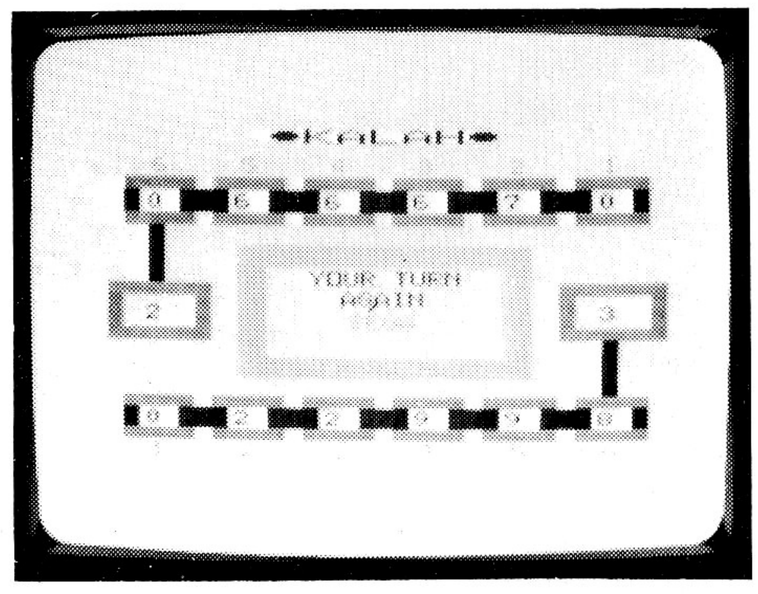
The classic maze chase game, eat the power pills move around the maze avoiding chasing ghosts until its your turn to give chase and score bonus points. A very fast all machine code program.

Laser Blast

One of the few real one or two player games available, you can pit your wits against the computer, which is intent on landing alien craft on the surface to destroy your ground installations you control, either an aerial gun sight, or a ground cannon as well as 3 smart bombs for the more dangerous moments.

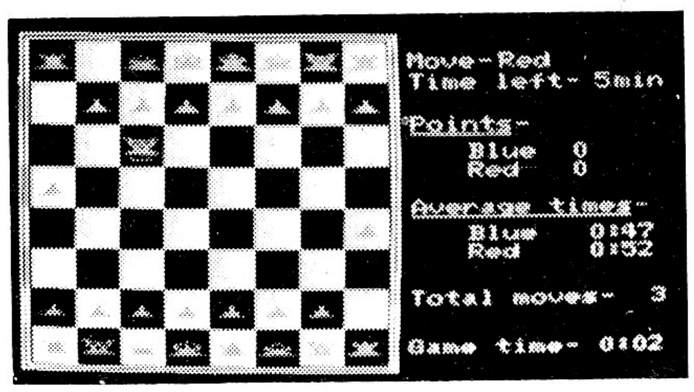
Kalah

This fascinating game originated over 3,000 years ago in Egypt and is still being played as a board game today all over the world. The game involves outwitting the computer by moving a number of stones around a board to increase your score and foil the computer at the same time.



Chess

This program enables your T.V. screen to be used as a Chessboard. Castling and enpassant moves are allowed, time limits can be set, points are allocated for pieces taken. Allowing you to improve your skill at this ever popular game (NOTE Game must be played by two players.)



Bugaloo

You were the unfortunate victim of circumstances as you were taken captive by the evil Gonzo race of aliens. They forced you to play their planet's survival game suspended in space.

You have to outsmart the Gonzos by jumping from platform to platform, leaving a trail of spots. When you have visited every platform, you have to start again, but it will be much harder. Every now and again a disk will appear and if you jump on it, 500 bonus points are awarded.

This one player game uses the SC3000's graphic and sound capabilities to their fullest extent.

File System

Allows you to enter up to about 300 pages of information about customers, or personal record keeping. This can be stored on tape and read adjusted, or transferred to paper via the printer.

Files can also be searched by the computer to give details of all files which contain certain pieces of information, ignoring those files which are not required.

Typing Tutor

Gives word and sentence tests to increase your typing skills. It montiors your words per minute performance and automatically displays the information as a bar chart every tenth test.

Cheque Book Reconciliation

Helps you check your monthly bank statement to ensure you pay only for what you have purchased. An invaluable home financial aid.

Loan & Mortgage Calculator

A financial package to assist you in making the right decisions when taking out a loan for homes, cars, etc. Perform "What If" calculations to ensure you pay the least amount of interest possible.

Graph & Chart Presentation

This enables the user to display figures and information quickly and easily in graphic form as either bar, line or pie charts. A frequently used easy to read display for presenting data.

Music Demo

Illustrates the capabilities of the excellent music cartridge, (which you require to run the cassette) and contains 10 pieces of music.

Accounts Receivable/Accounts Payable

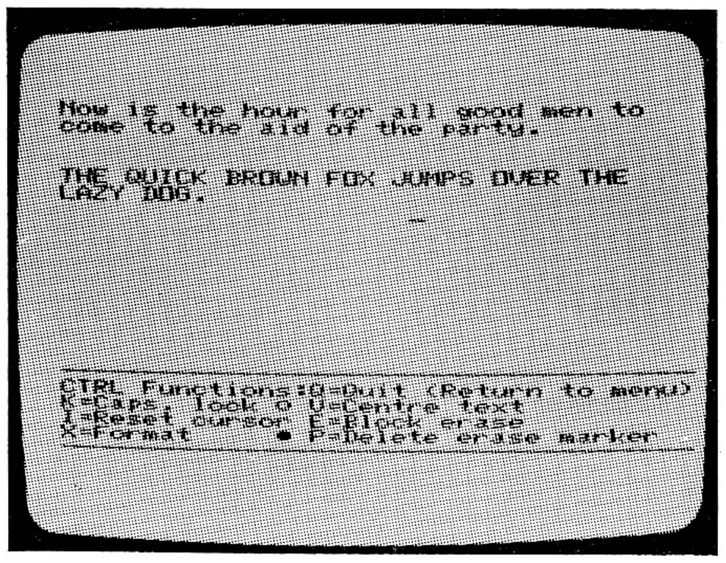
Two separate programs which run much the same with different relevant prompts, enabling you to keep accurate records of your daily/monthly accounts of debtors and creditors by storing the information on tape, and having the computer record regular totals.

Mailing List

Allows you to keep cassette records of addresses and produce printed lists of sections, or all of the stored addresses.

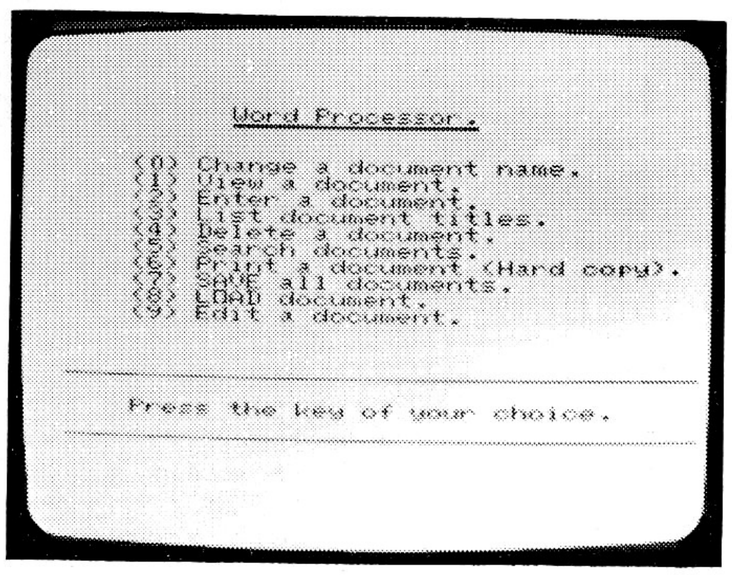
Sega Easy Writer

A Word Processing package which works on one letter at a time, automatically formatting the letters, with search and replace options, paragraph delete, automatic centering, forward and reverse scrolling. Each letter can be over 1,000 characters in length and will be printed out on the Sega Printer Plotter.


Now is the hour for all good men to
come to the aid of the party.
THE QUICK BROWN FOX JUMPS OVER THE
LAZY DOG.
CTRL Functions:0=Quit (Return to menu)
T=Caps lock 0 U=Centre text
I=Reset cursor E=Block erase
X=Format P=Delete erase marker

32K Word Processor

Much the same features as the easy writer with the ability to hold 99 pages of letters in memory. Can only be used with 32K Sega.


Word Processor.
(0) Change a document name.
(1) View a document.
(2) Enter a document.
(3) List document titles.
(4) Delete a document.
(5) Search documents.
(6) Print a document (Hard copy).
(7) SAVE all documents.
(8) LOAD document.
(9) Edit a document.
Press the key of your choice.

Sega Personal Record Keeper

Much the same package as the Word Processor with different prompts allowing you to store a variety of personal records on tape and also hard copy onto the Plotter Printer. Use with 32K.

Enterprise Escape

You are on the bridge of the deserted Starship Enterprise. The engines are out and the Enterprise in a decaying orbit. You must start the engines and save the ship, get off the ship, or die!

Mars Adventurer

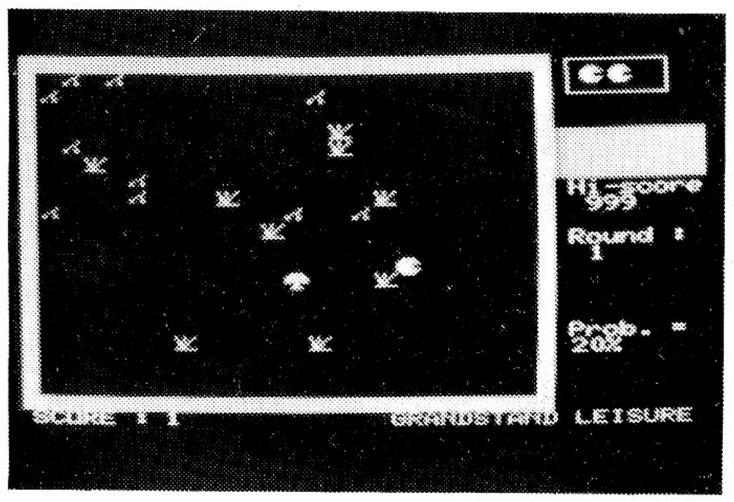
While on a routine space patrol you encounter a freak meteor storm and crash land on the planet Mars. You have to get home, but first you must explore the Martian habitat, deal with hostile Martians, and repair your ship, but it isn't easy!

Bit Byte

In this game you are restricted to a field surrounded by fences. Your existence depends on your skill at eating the grass in this field. However, some wild berries have started growing in your field. You have a very bad allergy to these berries that kill you.

The fences are electrified and when you touch them you are instantly fried. When you have successfully cleared all the grass from the field, move on to another field with some fences.

Occasionally a mushroom may appear which once eaten can give a bonus of 150-250 points. However, the mushroom may turn out to be a toadstool which will kill you!



Z80 Disassembler

This program allows you to look at any block of the computer's memory. You can work your way through both the Ram and the Rom obtaining a disassembled listing of the memory, either as ASCII and hexidecimal codes, or as machine code mnemonics.

The information can be directed to a Sega S.P. 400 Printer Plotter if required.

Death Satellite

You are the pilot of a small spaceship out for a 'sunday drive'. Unfortunately you forgot to fill the tank before you left the earth's atmosphere and you are now desperately low on fuel. There are no nuclear filling stations anywhere in the near vicinity and it looks as though you are going to be marooned in space.

Your only hope of survival is reaching the old deserted satellite. You hand on the last drop of fuel.

Your Mission:

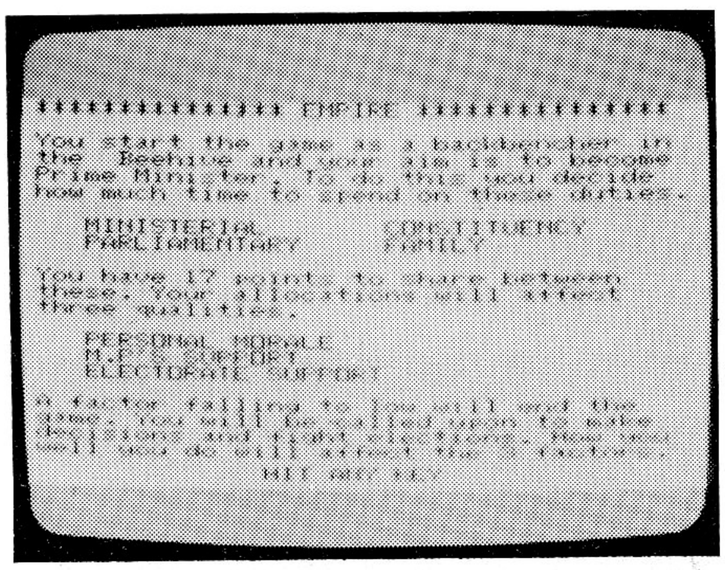
To seek out the radioactive fuel pellet and use it to refuel your ship.

A Word of Warning:

The satellite is full of many perils and potential death traps.

Empire

This game is essentially computerised politics. Each year you have a certain amount of time which you can divide between your ministerial duties, your constituency, Parliament, and your family. Your actions in these areas determine your voter support, your own personal morale and the amount of backing you have from your fellow M.P.'s. You will be called upon to make decisions and fight elections. Every so often you will have the opportunity to be promoted. You start off as a back-bencher and if you are successful rise through the ranks and become Prime Minister!


EMPIRE
You start the game as a backbencher in the Beehive and your aim is to become Prime Minister. To do this you decide how much time to spend on these duties.
MINISTERIAL
PARLIAMENTARY
CONSTITUENCY
FAMILY
You have 17 points to share between these. Your allocations will affect three qualities.
PERSONAL MORALE
M.P. SUPPORT
ELECTORATE SUPPORT
As for voting to you will see the vote, you will be called upon to make decisions and that also affect the qualities.
HIT ANY KEY

CHRISTCHURCH LOCAL USER CLUB

We have many requests for a Christchurch club to be formed, and request any interested persons send an address and contact Phone No if they wish to be put in touch with other users in this area: with a view to meetings being arranged.

Please Post to

Graham Rudman
29 Primrose St
CHRISTCHURCH 5

F BASIC

We have now received the production version of F Basic which is used on the Super Control Station. There are 33 new commands, statements and functions giving you full file and disk handling routines. The Disk Drive Basic has 124 basic keywords which greatly increases the power of the cartridge basic which only has 91 keywords.

F Basic is loaded from disk into the Ram of the system, when the unit is first turned on. You are able to open up to 8 files simultaneously, allowing you to access information from a while disk.

As the basic language is not 'on board' but on disk, other languages like Logo will be able to be loaded onto the Super Control Station from disk.

New commands are:

BOOT	Reloads F Basic from the disk.
CLOAD	Loads a program from cassette.
COMLOAD	Receives information/programs via the RS232C interface.
SAVE	Saves a program on cassette.
COMSAVE	Sends information/programs via the RS232C interface.
FILES	Displays the names of all programs and files on the disk.
LFILES	Prints the names of all programs and files on the disk.
MAXFILE	Sets the number of files that can be opened.
MERGE	This merges a program on disk with the program in memory.
NEWON	Sets the starting address for the basic programming area.
UTILITY	Enters the disk utility mode allowing new disks to be formatted, and copying of disks.

STATEMENTS

CLOADM	Loads machine code from cassette.
CLOSE	Closes specified files.
COMSET	Used for the RS232C interface. Sets datalength and parity.
CSAVEM	Saves machine code onto cassette.
DSKI$	Reads directly from disk.
DSKO$	Writes directly to disks.
GET	Read variable data from random files.
INPUT#	Accepts value or string from indicated sequential file or from the RS232C interface.
KILL	Erases a file or program from the disk.
LIMIT	Sets end address for the BASIC program area.
LOADM	Loads machine code programs into specified memory area.
NAME	Changes the names of programs on disk.
OPEN	Opens disk files.
PRINT#	Writes value of character strings to specified sequential file or RS232C interface.
PUT	Writes value of expressions to random file.
SAVEM	Save machine code programs onto disk.
VERIFYM	Compares machine language programs saved on cassette with programs in memory.

FUNCTIONS

DSKF	Used to check free space on disk.
EOF	Used to check whether sequential file is read to the end of the file, – 1: Yes, Ø: No
INPUT$	(x,#y) Sets string from sequential file #y to length x
LOC	Gives logical offset in file.
LOF	Gives file size.